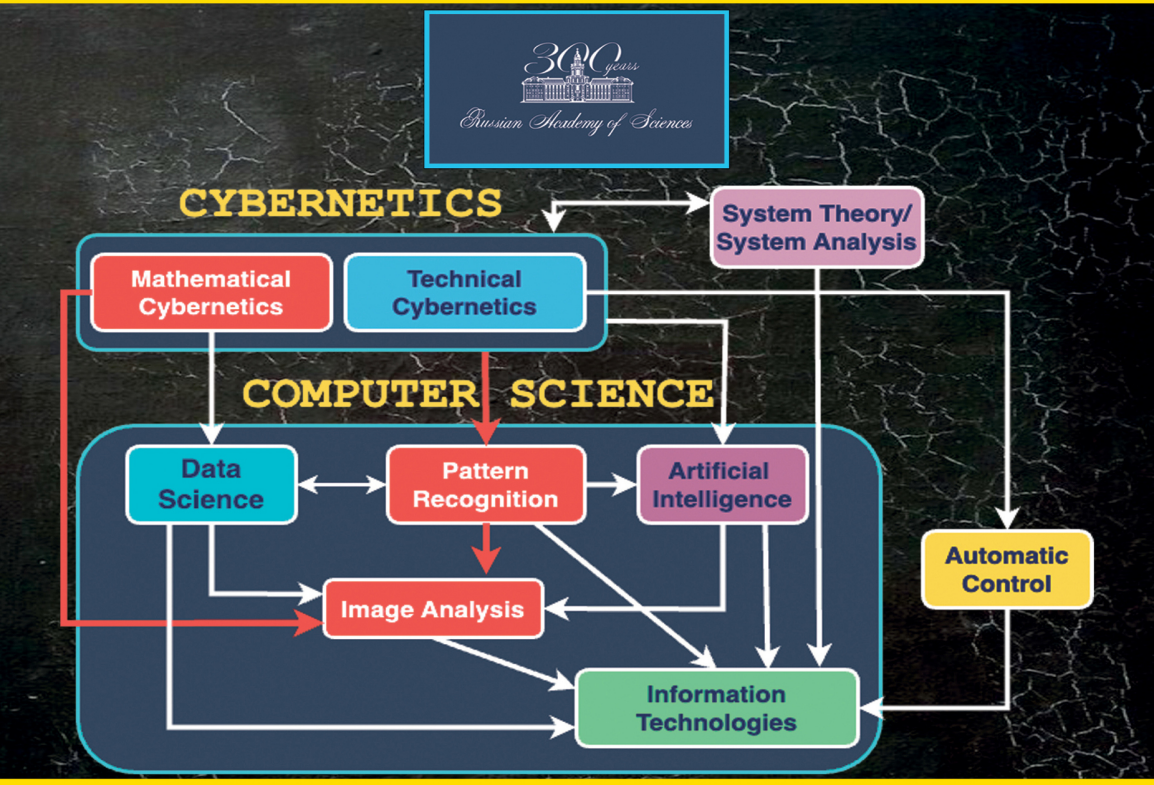
Series on Language Processing, Pattern Recognition, and Intelligent Systems — Vol. 7

Image Analysis and Pattern Recognition

State of the Art in the Russian Federation

Editors

Igor Gurevich • Vera Yashina



World Scientific

Image Analysis and Pattern Recognition

State of the Art in the Russian Federation

Series on Language Processing, Pattern Recognition, and Intelligent Systems

Print ISSN: 2661-4316
Online ISSN: 2661-4324

Co-Editors

Ching Y. Suen
Concordia University, Canada
parmidir@enes.concordia.ca

Lu Qin
The Hong Kong Polytechnic University, Hong Kong
csluqin@comp.polyu.edu.hk

Published

Vol. 7 *Image Analysis and Pattern Recognition: State of the Art in the Russian Federation*
edited by Igor Gurevich and Vera Yashina

Vol. 6 *Advances in Pattern Recognition and Artificial Intelligence*
edited by Nicola Nobile, Marleah Blom and Ching Y. Suen

Vol. 5 *Frontiers in Pattern Recognition and Artificial Intelligence*
edited by Marleah Blom, Nicola Nobile and Ching Y. Suen

Vol. 4 *Computational Linguistics, Speech and Image Processing for Arabic Language*
edited by Neamat El Gayar and Ching Y. Suen

Vol. 3 *Social Media Content Analysis: Natural Language Processing and Beyond*
edited by Kam-Fai Wong, Wei Gao, Wenjie Li and Ruifeng Xu

Vol. 2 *Advances in Chinese Document and Text Processing*
edited by Cheng-Lin Liu and Yue Lu

Vol. 1 *Digital Fonts and Reading*
edited by Mary C. Dyson and Ching Y. Suen

Series on Language Processing, Pattern Recognition,
and Intelligent Systems — Vol. 7

Image Analysis and Pattern Recognition

State of the Art in the Russian Federation

Editors

Igor Gurevich
Vera Yashina

Federal Research Centre "Computer Science and Control"
of the Russian Academy of Sciences, Moscow, Russian Federation

World Scientific

NEW JERSEY • LONDON • SINGAPORE • GENEVA • BEIJING • SHANGHAI • TAIPEI • CHENNAI

Published by

World Scientific Publishing Co. Pte. Ltd.

5 Toh Tuck Link, Singapore 596224

USA office: 27 Warren Street, Suite 401-402, Hackensack, NJ 07601

UK office: 57 Shelton Street, Covent Garden, London WC2H 9HE

Library of Congress Cataloging-in-Publication Data
Names: Gurevich, Igor, editor.
Title: Image analysis and pattern recognition : state of the art in the Russian Federation / editors, Igor Gurevich, Vera Yashina, Federal Research Centre "Computer Science and Control" of the Russian Academy of Sciences, Moscow, Russian Federation.
Description: [Hackensack] New Jersey : World Scientific, 2025. | Series: Series on language processing, pattern recognition, and intelligent systems, 2661-4316 ;vol. 7 | Includes bibliographical references and index.
Identifiers: LCCN 2024003617 | ISBN 9789811267208 (hardcover) | ISBN 9789811267215 (ebook for institutions) | ISBN 9789811267222 (ebook for individuals)
Subjects: LCSH: Pattern recognition systems--Russia (Federation) | Image analysis--Russia (Federation)
Classification: LCC TK7882.P3 I36 2025 | DDC 006.4--dc23/eng/20241009

British Library Cataloguing-in-Publication Data
A catalogue record for this book is available from the British Library.

Copyright © 2025 by World Scientific Publishing Co. Pte. Ltd.

All rights reserved. This book, or parts thereof, may not be reproduced in any form or by any means, electronic or mechanical, including photocopying, recording or any information storage and retrieval system now known or to be invented, without written permission from the publisher.

For photocopying of material in this volume, please pay a copying fee through the Copyright Clearance Center, Inc., 222 Rosewood Drive, Danvers, MA 01923, USA. In this case permission to photocopy is not required from the publisher.

For any available supplementary material, please visit
https://www.worldscientific.com/worldscibooks/10.1142/13167#t=suppl

Desk Editors: Soundararajan Raghuraman/Balasubramanian Shanmugam/Steven Patt

Typeset by Stallion Press
Email: enquiries@stallionpress.com

Printed in Singapore

© 2025 World Scientific Publishing Company
https://doi.org/10.1142/9789811267215_fmatter

About the Editors



Igor B. Gurevich was born in 1938. He earned his diploma in engineering in Automatic Control and Electrical Engineering in 1961 at the National Research University Moscow Power Engineering Institute, Moscow, USSR and doctorate in Mathematical Cybernetics in 1975 at the Moscow Institute of Physics and Technology (National Research University), Moscow, USSR. He is a Leading Researcher at the Federal Research Center "Computer Science and Control" of the Russian Academy of Sciences, Moscow, the Russian Federation. He has worked from 1960 till now as an engineer, researcher, and lecturer in industry, research institutions, medicine, and universities and from 1985 in the USSR/Russian Academy of Sciences. His area of expertise includes mathematical theory of image analysis, image-mining, image understanding, mathematical theory of pattern recognition, theoretical computer science, medical informatics, applications of pattern recognition and image analysis techniques in biology, medicine, automation of scientific research, and knowledge-based systems.

Gurevich suggested, proved, and developed with his pupils Descriptive Approach to Image Analysis and Recognition (DAIA). In DAIA framework, a new class of image algebras was introduced, defined, and studied (Descriptive Image Algebras); new types of image models were introduced, classified, and studied; axioms of Descriptive Theory of Image Analysis were introduced; a common

model of image recognition process was defined and studied; new settings of image analysis and recognition problems were introduced; a notion "image equivalence" was introduced and studied; new classes of image recognition algorithms were defined and studied; an image formalization space was introduced, defined, and studied.

Listed results were used in the development of software kits for image analysis and recognition and for the solution of important and difficult applied problems of automated bio-medical image analysis.

Gurevich is an author of two monographs and of 344 papers in peer-reviewed journals and conference proceedings indexed in Web of Science, Scopus, and Russian Science Citation Index on the platform of Web of Science, and in many other bibliometrics databases; 34 invited papers at international conferences, and a holder of eight patents.

He is Vice-Chairman of the National Committee for Pattern Recognition and Image Analysis of the Russian Academy of Sciences, Member of the International Association for Pattern Recognition (IAPR) Governing Board (representative from the Russian Academy of Sciences), and IAPR Fellow. He has been the PI of 63 R&D projects as part of national and international research programs. He is Vice-Editor-in-Chief of the *Pattern Recognition and Image Analysis: Advances in Mathematical Theory and Applications* (international journal of the RAS), member of editorial boards of several international peer-reviewed scientific journals, and member of the program and technical committees of many international scientific conferences.

His teaching experience includes the following: Moscow Lomonosov State University, RF (assistant professor), Dresden Technical University, Germany (visiting professor), George Mason University, USA (research fellow). He was supervisor of 6 PhD students and many graduate and master students (Moscow Lomonosov State University, RF; Federal Research Center "Computer Science and Control" of the Russian Academy of Sciences, Moscow, the Russian Federation.



Vera V. Yashina was born in 1980. She earned a diploma in mathematics at Moscow Lomonosov State University (2002) and doctorate in Theoretical Foundations of Informatics in 2009 at Dorodnicyn Computing Center of the Russian Academy of Sciences, Moscow. She is leading researcher at the Department of "Mathematical Problems of Information Recognition and Analysis" at the Federal Research Center "Computer Science and Control" of the Russian Academy of Sciences, Moscow, the Russian Federation. She has worked from 2001 until now in the Russian Academy of Sciences. Her scientific expertise includes mathematical theory of image analysis, image algebras, models, and medical informatics.

The main results were obtained in the mathematical theory of image analysis together with I.B. Gurevich: (1) New mathematical objects are introduced and investigated: the space of image formalization, descriptive image algebras, descriptive generating trees, descriptive algorithmic schemes; (2) descriptive models of image analysis are defined and investigated: image models, image transformation models, models of generation of descriptive algorithmic schemes. A number of automated software systems have been developed. The axioms of descriptive image analysis are proposed.

She is scientific secretary of the National Committee for Pattern Recognition and Image Analysis of the Russian Academy of Sciences. She is a member of the Educational and Membership Committees of the International Association for Pattern Recognition. She is a Chair of Technical Committee No. 16 "Algebraic and Discrete Mathematical Techniques in Pattern Recognition and Image Analysis" of the International Association for Pattern Recognition. She has been a member of many R&D projects as part of national and international research programs. She is a member of the editorial board of *Pattern Recognition and Image Analysis. Advances in Mathematical Theory and Applications* (international journal of the RAS). She is the author of 113 papers in peer-reviewed journals, conference, and workshop proceedings and of 10 invited papers at international conferences and a holder of five patents. Her teaching experience is at Moscow State Lomonosov University, RF. She was supervisor of several graduate and master students.

This page intentionally left blank

© 2025 World Scientific Publishing Company
https://doi.org/10.1142/9789811267215_fmatter

Image Analysis and Pattern Recognition in the Russian Federation: Main Results and Development Trends

Igor B. Gurevich* **and Vera V. Yashina†**

Federal Research Center "Computer Science and Control" of the Russian Academy of Sciences, 44/2, Vavilov str., Moscow, the Russian Federation

**igourevi@ccas.ru*

†*werayashina@gmail.com*

The aim of presenting this monograph to readers — scientists, researchers, engineers, specialists, masters, and senior students specializing in computer science and applied mathematics — is to familiarize them with some aspects of the current state of fundamental and applied research in image analysis and pattern recognition conducted at institutes of the Russian Academy of Sciences and leading Russian universities. All chapters were written by foremost scientists and specialists of the Russian Federation directly involved in fundamental and applied research in image analysis and pattern recognition, i.e. first-hand information directly from the cutting edge of research.

Keywords: Image analysis, pattern recognition, mathematical theory of image analysis, mathematical theory of pattern recognition, artificial intelligence, scientific schools in image analysis, scientific schools in pattern recognition.

1. General Characteristics of Directions in Image Analysis and Pattern Recognition

Image analysis (IA) and pattern recognition (PR), as independent scientific directions, originated and underwent initial development within the framework of cybernetics (mainly, mathematical cybernetics), then, as cybernetics evolved into computer science, they became leading directions of the latter.

As is known, computer science is defined as the science of methods and processes of collecting, storing, processing, analyzing, and evaluating information, ensuring its possible use for decision-making. IA and PR fully comply with this definition with the refinement that the information used in these leading branches of computer science has very specific properties that significantly complicate its processing and analysis. Other distinctive features of IA and PR are functional problems, the solution of which is the raison d'être of these scientific directions, namely, formulation of problems, their content, and classes of admissible solutions.

In this sense, the IA and PR objects are as follows:

(1) source information — aggregate data available for use,
(2) implementations and information models — the aggregate types of representation of information: "recorded information" (symbols, signals, images, texts) and symbolic (mathematical) models obtained from applying preprocessing methods to the source information, ensuring its reduction to a form that allows the use of mathematical methods for information processing, analysis, and evaluation,
(3) mathematical methods for transforming implementations, models, and the source information itself,
(4) implementation of mathematical methods for converting implementations, models, and actual source information in the form of computer programs,
(5) the results of processing, analyzing, and evaluating information containing both decisions and processed information necessary for their adoption.

IA and PR as fundamental scientific directions are related to mathematics through mathematical models, mathematical logic, and the

theory of algorithms; to linguistics through formal languages and sign systems; and to information theory through procedures for assessing the efficiency of algorithms and information representations.

IA and PR began to take shape in the 1960s as scientific directions in dual-use information technologies in connection with rise of complex large-scale applied problems that cannot be analyzed by methods of classical mathematics (space research, development of atomic and nuclear weapons, strategic and tactical planning of military operations, analysis of heterogeneous intelligence information, and meteorological prediction). These problems, to one degree or another, required a description of the processes being studied in terms of collecting, storing, processing, analyzing, and evaluating information and solutions in the form of an integral assessment of objects, processes, situations, and natural and man-made phenomena (class/type of object, process, and phenomenon), and administrative, preventive, or prognostic decisions.

The mathematical apparatus of IA and PR includes their mathematical principles, computation theory, algorithms, and data structures. IA and PR use mathematical logic: algebraic logic, discrete mathematics, logical conclusions, formal evidence, number theory, graph theory, type theory, category theory, computational geometry (the study of algorithms for solving problems formulated in geometric terms), and other branches of classical mathematics. An important role in IA and PR is played by computation theory, which includes the automata theory (the study of logical structures for solving problems), computability theory (establishing what can and cannot be calculated, including on modern computers), computational complexity theory (fundamental restrictions, primarily on time and memory size for classes of computational methods), and quantum computing theory (representation and transformation of data using quantum properties of particles and quantum mechanisms). Algorithms and data structures are the primary tools and objects for studying IA and RE. This section includes the algorithms themselves (logical formalization of computational processes and their efficiency) and analysis of algorithms (temporal and spatial complexity of algorithms) and data structures (organization and rules for data transformation).

Relationships of IA and PR with mathematics are not limited to the mere use of mathematical methods. Mathematics and IA and

PR also share common research objects. Thus, algorithms that are the object of study in the mathematical theory of algorithms can be considered at the same time as some formalized systems and serve for IA and PR not only as a tool but also as an object of research. The approach to studying this object and, therefore, problems arising in mathematics and IA and PR, however, differ significantly. For mathematics, an algorithm acts primarily as one of the fundamental concepts of the foundations of mathematics. Therefore, the main problem is to study the general properties of this concept, for which it is necessary to reduce its definition to a minimum number of the simplest fundamental concepts and operations. IA and PR (as well as applied mathematics) pose their problem as the development of practically convenient methods for synthesizing specific systems of mathematical objects, including algorithms. This practical orientation makes it necessary to develop procedural and problem-oriented algorithmic languages that are fairly convenient for use. Instead of the interest characteristic of mathematics in the fundamental possibility of establishing equivalence in certain classes of algorithms, IA and PR are primarily interested in creating an apparatus convenient for actually performing equivalent transformations of algorithms. Instead of the simplest form of representing information as words in an abstract alphabet, IA and PR study the complex data structures necessary for efficient implementation of algorithms on a computer. A similar difference in approaches is revealed when studying other mathematical objects in IA and PR (abstract automata, logical networks, etc.).

The methodological basis of IA and PR consists of basic computer science methods: algorithmization, formalization, and mathematical modeling.

Algorithmization, which originated in mathematical cybernetics, is one of the most important tools of IA and PR. The algorithmization method follows the general principle of studying objects in terms of information and operations on information and comes down to an algorithmic description and specification of information processes. Algorithmization is the compilation of a mathematical description (mathematical model) of the object, process, and phenomenon being studied, i.e. the implementation of cause-and-effect relationships and other patterns as a directed process of information processing according to formal rules. The source of initial information

for algorithmization is theoretical and experimental data, as well as heuristic, informal information about the process being studied. Algorithmization of information processes involves their precise description in precisely constructed artificial languages: information-logical and algorithmic, i.e. artificial languages designed to represent the corresponding algorithms. The theory of algorithms (recursive function theory, Turing machines, and Post combinatorial processes) began to develop in the 1930s, defining computation models and formal designs for general-purpose calculators before the advent of computers. The algorithmic procedures generated by these models turned out to be machine-oriented and formed the basis for organizing computer calculations and developing modern algorithmic languages.

Formalization, which is a necessary condition for algorithmization (in particular, through the construction of formalized languages, thesauri, and ontologies to represent some content area), and algorithmization (representation of some information processing process in the form of a deterministic prescription: an algorithm) require rigorous concepts.

Another difficulty of formalization and algorithmization encountered in IA and PR is associated with refining the meaning of expressions in scientific and natural languages: semantic issues. It is believed that such a refinement of semantics is promising, based on algorithmic methods, and the operation of abstract concepts is replaced by the operation of symbols as constructive material objects: signs. Such refinement of semantics is a necessary condition to automate intellectual activity, which undoubtedly includes IA and PR.

These difficulties of formalization significantly complicate the use of IA and PR in a number of natural sciences (biology, geology, medicine, and others), especially in the humanities.

Mathematical modeling is one of the most important IA and PR tools for obtaining knowledge and solving problems as a result of information processing. This method replaces original objects with their mathematical models and conducts computational experiments with them, as a result of which the sought characteristics and properties of objects, processes, and phenomena are determined. Mathematical modeling, or more broadly, information modeling of any object consists of the development and use of the model–algorithm–program triad. Its construction puts a tool in the hands of the researcher, which is first debugged and tested by comparing the

results obtained in trial computational experiments with the results of full-scale experiments. After establishing the adequacy of the triad to the original object, computational experiments are carried out, which yield all the required knowledge and data.

IA and PR as a scientific and technical direction arose and formed from the need to solve problems of analysis and evaluation of incomplete, contradictory, complexly structured, semantically rich, difficult-to-formalize, heterogeneous, and noisy information using computationally efficient mathematical methods. The initial information in these problems is numerical, symbolic and expert information, images, speech, signals, texts, documents, drawings, and arbitrary combinations of these types of initial data.

Methods and tools created and developed within this scientific and technical direction are designed to solve applied intellectual decision-making, diagnostic, identification, and prediction problems. Typical examples of problems in this area are decision-making and prediction problems that arise, in particular, in technical diagnostics, non-destructive testing, remote sensing, environmental monitoring, prediction, and diagnostics in medicine, planning, search in geology, prediction in chemistry, and automation of scientific research (event detection, search, recovery, and inference of empirical patterns). A distinctive and fundamental feature of these problems is the impossibility of using classical mathematical (analytical) models to formalize and represent source data, which can be specified by numerical arrays, images, signals, texts, verbal descriptions, or a hybrid of these methods for presenting information. The information in these problems is "bad" by definition: characteristic features are incompleteness, inconsistency, poor structure, lack of formalization, and vagueness. The solution to these problems is based not on purely computational models but on models based on procedures for analyzing and evaluating information about the problem — mainly precedents, indirect characteristics available for measurement, logical and physical restrictions, and contextual and implicit knowledge.

The mathematical formulation of problems of pattern recognition, prediction, analysis, and understanding of images and signals does not depend on the subject area and specific type of object of analysis.

PR methods constitute the basis of information analysis and support, which is the main function of modern computer technology in industry, scientific research, economics, and the social sphere.

Since the 1960s, most pattern recognition applications have been related to IA. The role of the image as an object of analysis and evaluation is determined by its specific and integral information properties. A key area of IA is the automation of extracting knowledge and data from images (image mining) necessary for making intelligent decisions. The development of the mathematical theory of IA follows the path of its "algebraization" based on the algebraic approach to recognition and classification problems and its specialization in the case of representing initial information in the form of images — the descriptive theory of IA, as well as the development of image algebras, including descriptive image algebras. The difficulties in solving IA problems are determined by the need to ensure some balance between very contradictory factors: the goals of analysis, the nature of human visual perception, methods and means of recording, generation and storage of images, and acceptable mathematical and computational tools for analysis. The main contradiction is related to the "pictorial nature" of the image itself and the formal ("symbolic") methods of its representation and analysis.

IA in a broad sense includes all types of image transformation: processing, image analysis in the narrow sense, recognition, and understanding. Image processing is the transformation of one image into another using compression, reconstruction, visual enhancement, quantization, and filtering. Image analysis in a narrow sense ensures reduction of an image to a form convenient for recognition, i.e. construction of formal representations and image models.

The main tasks of image analysis in a narrow sense are segmentation, extraction of features that describe the composition and content of the image, calculation of characteristics used to build an image model, and model synthesis.

Image recognition consists of formulating and solving PR problems when specifying the initial information in the form of images and models thereof.

As a result, a recognized image, its fragments, or individual objects represented on it are attributed to a certain class or are divided into disjoint subsets.

Understanding images is interpreted as an imitation of the functional capabilities of the human visual system, in particular, as the extraction of information about three-dimensional objects from two-dimensional images.

The result is a formal description of the image in the language of its properties, elements, and relationships between them, which is suitable for making intelligent decisions about the three-dimensional world.

Understanding images is often equated with the analysis of scenes and is considered as the main problem of automated machine vision systems.

IA uses a wide range of mathematical methods of algebra, geometry, discrete mathematics and mathematical logic, probability theory and mathematical statistics, mathematical analysis, as well as methods of the mathematical theory of PR, signal processing theory, physics, and optics.

2. Development of Image Analysis and Pattern Recognition in the Russian Academy of Sciences

Problems of intellectual decision-making based on data analysis require using some formal representation of original information — ideally, a mathematical model.

Research in the field of IA and PR began to develop intensively in the USSR Academy of Sciences in the mid-1950s, as a result of which, in the 1960s–1970s, completely original approaches to recognition were created based on the traditions of the Russian mathematical school and the latest results in the field of discrete mathematics, mathematical cybernetics, and probability theory. These approaches gradually began to prevail in Soviet theoretical and applied computer science. In the development, research, and implementation of new mathematical IA and PR methods, powerful original domestic scientific schools arose, which is traditionally a characteristic feature of the USSR Academy of Sciences.

The main goal of these studies was and continues to be the development of general mathematical methods for solving classes of problems, in contrast to the widely practiced and hitherto widespread approach of searching for a particular solution method for each individual problem.

As a result of this choice of the main fundamental research directions, the scientific schools of the Russian Academy of Sciences made

an extremely significant contribution to the development of the mathematical theory of IA and the mathematical theory of PR.

In the mid-1950s, a new scientific direction began to take shape in cybernetics, related to the development of the theoretical foundations and practical implementation of devices and systems designed to recognize unknown objects, events, phenomena, and processes. This direction is called "pattern recognition." The first civilian applications were devoted to problems in astronomy, nuclear physics, biophysics, radiophysics, and character recognition.

During this "classical" period until the end of the 1960s, the development of IA and PR was spontaneous, although back in the 1940s, Academicians of the USSR Academy of Sciences A.N. Kolmogorov and A.Ya. Khinchin posed the problem of separating a mixture of two distributions, which can be considered the first mathematical formulation of the PR problem in the USSR.

Nevertheless, some significant results were obtained during this period.

In 1954 at Moscow State University, the seminar "Automata and Thinking" began under the supervision of Academician of the USSR Academy of Sciences A.A. Lyapunov, one of the founders of cybernetics in the USSR. This seminar was attended by physiologists, linguists, psychologists, and mathematicians. It is generally accepted that it was at this time that artificial intelligence was born in the USSR.

One of the first studies in the field of radio research was by Academician of the USSR Academy of Sciences A.A. Kharkevich. Many Soviet scientists made a significant contribution to the development of this area. Among them were M.A. Aizerman, V.M. Akhutin, M.M. Bongard, V.N. Vapnik, V.M. Glushkov, Yu.I. Zhuravlev, N.G. Zagoruiko, A.G. Ivakhnenko, V.A. Kovalevsky, V.S. Fain, A.Ya. Chervonenkis, and others.

Among the most significant results obtained by domestic scientists in the 1960s, the "Kora" algorithm by M.M. Bongard is noteworthy, which models the activity of the human brain during PR. A great contribution to the domestic school of artificial intelligence (AI) was made by prominent scientists M.L. Tseitlin, V. Ya. Pushkin, and M.A. Gavrilov, whose students were pioneers of this science in the USSR (e.g. the well-known Gavrilov school).

In the 1960s, in-depth research in the field of radioactivity began under the supervision of Academician of the Russian Academy of Sciences Yu.I. Zhuravlev. During 1976–1978, Zhuravlev published a series of works on the algebraic approach to recognition, classification, and prediction problems, which soon became widely known, in particular, the problem of synthesizing correct recognition algorithms. These works determined the state of the art of the entire problematic of the mathematical theory of PR and many related areas of applied mathematics and computer science, which gave rise to a stream of research on the algebraization of PR and IA, which to a large extent determine the global leadership of Zhuravlev's scientific school in mathematical PR methods.

In 1961, the Institute of Information Transmission Problems of the USSR Academy of Sciences was founded, where the Laboratory of Iconics was created. From the very beginning, the scope of the laboratory was oriented toward solving not only current practical but also general theoretical problems related to video information processing. At the Laboratory of Iconics, many pioneering research and developments were carried out in the field of creating mathematical models and methods of digital image processing, which became the basis for many scientific studies in the field of video information conversion. On the basis of the Laboratory of Iconics, a scientific school on digital IA was formed, which for a long time occupied a prominent place in the country. Such areas of research as image models, image analysis and restoration, digital holography, processing of space images of the Earth and planets, computer vision, video signal compression, and others have emerged.

An extremely significant contribution to the development of PR theory was made by research conducted at the oldest Soviet scientific organization in the field of automation, cybernetics, and computer science: the Institute of Control Problems (Automation and Telemechanics) of the USSR Academy of Sciences (Moscow). It created new methods for analyzing complexly organized data:

(a) potential functions method,
(b) theory of learning and self-learning systems,
(c) generalized portrait method,
(d) reconstruction of dependences based on empirical data,
(e) automatic classification methods,
(f) expert classification analysis

(M.A. Aizerman, E.M. Braverman, L.I. Rozonoer, Ya.Z. Tsypkin, V.N. Vapnik, A.Ya. Chervonenkis, I.B. Muchnik, and A.A. Dorofeyuk).

In 1983 in the Scientific Council on the complex problem of "Cybernetics" of the USSR Academy of Sciences (SCC AN USSR), the Laboratory of Cybernetic Methods in Informatics (LCMI) was organized.

Fundamental research carried out at LCMI under the direction of Dr. Eng. I.B. Gurevich was completely devoted to constructing the mathematical theory of IA. In 1985, Gurevich proposed the specialization of Zhuravlev's "Algebraic approach to solving problems of recognition, classification, and prediction" to the case of representing source data in the form of images and other poorly structured information.

The proposed specialization was based on the prospects of using unified algebraic methods developed in the mathematical theory of PR to construct correct algorithms with respect to a specific recognition problem with numerical initial data when selecting and synthesizing image recognition algorithms.

In 1989, Gurevich, in the development of the indicated specialization of the algebraic approach, proposed and substantiated the Descriptive Approach to Image Analysis and Recognition (DA). Since then, the development of DA and the implementation of its principles and methods within the framework of the descriptive theory of IA have become the main topics of fundamental research at LCMI.

Research on the mathematical foundations of image analysis and recognition, carried out at SSC RAS in 1989–2006 was continued at the Computing Center of the Russian Academy of Sciences in 2005–2015, and from 2015 to the present, at the Federal Research Center Informatics and Control, the Russian Academy of Sciences (FRC IC RAS), Moscow, Russian Federation.

The current state and trends in the development of the mathematical foundations of image analysis and recognition are determined by the level of development of methods, models, and other results of DA studied and developed by Gurevich and his students; since 2002, a significant contribution to the descriptive theory of IA was made by V.V. Yashina (FRC IC RAS, Moscow, RF).

The most significant fundamental results, which were ahead of the world level of studies in these areas of computer science and

determined breakthrough areas of research for a long time, often to this day were obtained at the Computing Center of the Russian Academy of Sciences (Moscow), the Institute of Control Problems (automation and telemechanics of the Russian Academy of Sciences (Moscow), Institute of Information Transmission Problems RAS (Moscow, RF), Institute of Mathematics SB RAS (Novosibirsk, RF), Institute of Cybernetics NAS of Ukraine (Kyiv, Ukraine), Institute of Technical Cybernetics of the National Academy of Sciences of the Republic of Belarus (Minsk, Republic of Belarus), and Institute of Computational Mathematics and Mathematical Geophysics SB RAS (Novosibirsk).

The fundamental and most striking achievements of IA and PR obtained at the USSR Academy of Sciences and Russian Academy of Sciences include the following:

1. Algebraic approach to solving recognition and prediction classification problems based on Zhuravlev's algebra (Yu.I. Zhuravlev, Computing Center RAS (Moscow, RF)).
2. Analysis of complex data:
   (a) potential functions method,
   (b) theory of learning and self-learning systems,
   (c) generalized portrait method and reconstruction of dependencies based on empirical data,
   (d) automatic classification methods and expert classification analysis (M.A. Aizerman, E.M. Braverman, L.I. Rozonoer, Ya.Z. Tsypkin, V.N. Vapnik, A.Ya. Chervonenkis, I.B. Muchnik, and A.A. Dorofeyuk, Institute of Control Problems (Automation and Telemechanics RAS (Moscow, RF)).
3. The committee method, collective methods of machine learning based on majority generalized solutions of improper systems of algebraic equations, and inequalities (V.D. Mazurov, Ural School of Pattern Recognition; N.N. Krasovskii Institute of Mathematics and Mechanics, Ural Branch, Russian Academy of Sciences, Yekaterinburg, RF).
4. Processing of optical and digital images as information transmission tools (Institute of Information Transmission Problems,

Russian Academy of Sciences (Laboratory of Iconics; Laboratory of Digital Optics), Moscow, RF).

5. Two-dimensional grammars (M.I. Shlesinger, Institute of Cybernetics of the National Academy of Sciences of Ukraine, Kyiv, Ukraine).
6. Information technologies for creating digital maps (automation of the processes of input and recognition of cartographic images, object-oriented data structures) (S.V. Ablameyko, Institute of Technical Cybernetics of the National Academy of Sciences of the Republic of Belarus, Minsk, Republic of Belarus).
7. Information technologies for the development of geographic information systems (Yu.G. Vasin, Research Institute of Applied Mathematics and Cybernetics, Nizhny Novgorod State University, Nizhny Novgorod, RF).
8. Processing, analysis, and recognition of images and optical signals. The theoretical foundations for processing large hyperspectral data from remote sensing of the Earth have been developed; mathematical models, methods, and information technologies for intelligent analysis of optical biomedical images have been developed (V.A. Soifer, Samara Scientific School of Computer Image Processing, Institute of Image Processing Systems of the Russian Academy of Sciences, Samara, RF; Samara State Aerospace University, Samara, RF).
9. Collective decision rules (L.A. Rasstrigin, R. Erenstein, Institute of Electronics and Computer Science of the Academy of Sciences of the Latvian SSR, Riga, Latvia).
10. Method of group accounting of arguments (MGUA) (A.G. Ivakhnenko, Institute of Cybernetics of the National Academy of Sciences of Ukraine, Kyiv, Ukraine).
11. The descriptive approach to image analysis (I.B. Gurevich and V.V. Yashina, SCC RAS, Moscow, RF).

Unfortunately, for many well-known reasons, contacts between scientific schools of the USSR Academy of Sciences and the Russian Academy of Sciences and foreign colleagues were random and very limited. Therefore, both classical and modern results obtained by Russian researchers were virtually unknown to foreign scientists or were obtained by them independently and with some lag.

The main classes of theoretical recognition problems were identified:

(1) selection and identification of features that characterize objects and situations submitted for recognition, and the formation of a feature space,
(2) construction of models of objects and situations entered for recognition,
(3) solution of recognition problems,
(4) solution of prediction problems (training and recognition of dynamic objects and situations),
(5) decision-making based on the results of solving recognition and prediction problems.

The structure of the problem space is, in principle, similar for problems of recognition of numerical data, images, signals, and speech.

The objects of the mathematical theory of recognition are as follows:

(1) algorithms:
    (a) classical general-purpose recognition algorithms,
    (b) algorithms for processing, analysis, and recognition of images/spatial information,
    (c) specialized recognition algorithms,
(2) models of recognition algorithms:
    (a) algebraic models,
    (b) combinatorial-logical models,
    (c) statistical models,
    (d) descriptive models,
    (e) descriptive algorithmic schemes (DAS),
    (f) hierarchical multilevel classifiers,
    (g) multialgorithmic classifiers:
        - collective decision rules,
        - committee algorithms,
        - multialgorithmic classifiers based on the apparatus of image algebras,
        - neural network algorithms,
(3) optimization in spaces of algorithm models:
    (a) correction of recognition algorithms,

(b) algebraic methods,
(c) automation of DAS generation,
(d) optimization of multialgorithmic classifiers based on the selection of areas of expertise of algorithms.

The strategic goal of fundamental and applied research in the RF in the field of IA and PR is defined as automation of the synthesis of information technologies for recognition, analysis and understanding of images, the development of mathematical and computational methods and software packages that provide and support its implementation, as well as the development and implementation of an information technology system for solving classes, especially important and difficult individual applied problems.

This goal was achieved by solving the following main problems:

(1) creation of a unified methodological complex of mathematical, information methods and tools that provide the following:
   (a) automation, regularization, and optimization of the choice of algorithmic and software when solving applied PR, IA, and understanding problems,
   (b) automation of the development, research, and implementation of automated computing systems designed to transform and evaluate information represented as images, signals, alphanumeric data sets, and combinations thereof,
(2) creation of complexes of instrumental and applied software for the implementation of information technologies based on systems that include computational areas and algorithmic knowledge bases.

In accordance with these goals, work within this scientific direction of computer science is divided into the following groups:

(1) development, research, modification, and systematization of mathematical methods forming the basis of the algorithmic content of information technologies,
(2) development of methods for automating the synthesis and testing of information technologies,
(3) development of specialized information technologies (for classes of problems and subject areas),

(4) development and implementation of broad-purpose tools, databases, and knowledge bases to support PR, IA, and signal processing information technologies,
(5) development, adaptation, and implementation of methods for solving individual non-standard and particularly important PR and IA problems.

All of these groups of studies are closely related. The first three groups of studies involve fundamental and applied research and direct implementation of the results. The second two groups of studies require applied research and development. Essential components of the research part are studies on mathematical methods for ensuring algorithmic content, manufacturability, and functional content of algorithmic knowledge bases. The essential components of the part associated with software implementation are studies on the development of specialized interfaces and shells for working with the corresponding types of source data (alphanumeric arrays, images, speech, and signals) and the development of systems and software systems for the main standard platforms.

All research conducted in the field of IA and PR can be divided into the following areas:

I. Development of intelligent algorithmic and software based on modern mathematical methods for solving problems of recognition and classification of images and prediction in order to detect critical and non-standard objects, situations, and events in technology, economics, and natural science.

The goal is to develop intelligent algorithmic and software for solving problems of detecting critical and non-standard situations in technology, economics, and natural science based on modern mathematical methods of recognition, classification, and prediction.

The subject of the research was the theoretical problems of automatic and automated synthesis of high-precision recognition and prediction algorithms, focused on detecting rare situations and based on the combination of logical, algebraic, optimization, iterative, and other approaches for extracting and using knowledge and data using non-representative samples of precedents.

Main results:

(1) algorithms and software for solving a set of problems of automatic and automated analysis of various types of data, search for patterns, recognition, classification, and prediction,
(2) intelligent software systems for broad purposes for detecting critical and non-standard situations using poorly structured, unrepresentative, contradictory, and inhomogeneous data,
(3) tools for adapting these systems to various subject areas in technology, economics, and natural sciences.

II. Development of a basic complex of algorithmic content and software and information technologies based on modern mathematical methods for processing, analyzing, and evaluating information represented in the form of images.

The aim is to automate the development, research, and testing of image processing, analysis, and recognition algorithms; develop methods for creating and transferring technologies for extracting information from images; develop standard solutions for image analysis and evaluation systems; develop and implement methods for solving individual non-standard and particularly important analysis and image recognition problems.

The subject of the research was as follows:

(1) development and research of methods for modeling and representing images in problems of analysis and recognition,
(2) development and research of algorithms for converting and evaluating information in image recognition problems,
(3) development and research of image algebras,
(4) development and research of automation methods:
    (a) choice of a method for solving the given problem,
    (b) issuing of recommendations on the choice of solution method for the class of problems to which the given problem is attributed,
    (c) synthesis of algorithmic procedures for solving the presented problem,
    (d) issuing of recommendations for the synthesis of algorithmic procedures for solving the given problem,

(5) development of systems to automate the development, research, and testing of image processing, analysis, and recognition algorithms:
   (a) development and implementation of functional content of systems based on effective classes of image analysis and recognition algorithms,
   (b) development and implementation of a hierarchy of classes representing the main objects and procedures for analyzing and evaluating images,
   (c) development of integrated software environments to ensure the functioning of automation systems and tools for the development and application of algorithms for analysis and image recognition in multitasking mode, visualization, and editing of used data,
   (d) development of functional cores and basic software modules of automation systems,

(6) development of information technologies for extracting information from images.

Main results:

(1) basic technology for automating information extraction from images,
(2) the automated "Black Square" system supporting technology for automating information extraction from images — a system for automating the selection and development of methods and algorithms for image processing and analysis with the following main characteristics:
   (a) composition of the system — a database of algorithms, a tool environment for their development and use in multitasking mode, tools for visualizing and editing the data used, and an electronic multilevel help system on the system,
   (b) the functioning of the system is ensured by a single integrated software environment, built on a common information and methodological base and having a common navigation shell,
   (c) Windows OS,

(d) system functionality:

- solving basic image processing, analysis, and recognition problems in automatic mode,
- intellectual, informational, and lexicographical support,
- openness regarding objects processed by the system, libraries of algorithms, filling with thematic, contextual, lexicographic, and expert information,
- functional completeness of algorithmic content for problems within the system's area of expertise,
- the ability to modernize the system by including new and excluding outdated modules.

III. Development of broad-purpose tools, hardware, and software for creating and supporting applied intelligent systems for automating data analysis, diagnostics, monitoring, detection, and prediction.

The goal is to develop tools for working with semantically structured data (SSD), images, signals of various nature, dynamic objects, and images of the real world and virtual reality.

Subject of the research:

(1) development of tools for efficient representation, storage and transmission, access control, and protection of SSD in modern distributed information systems,
(2) development of tools for analysis, synthesis, comparison, and integration of real SSD and objects of the virtual world,
(3) development of instrumental software and algorithmic tools to support information technologies for processing, analysis, and detection of observed objects under conditions of *a priori* uncertainty,
(4) development of promising information technologies for the creation and support of biotechnical and medical intelligent diagnostic, monitoring, and prediction systems based on automatic analysis of biomedical signals and images with a complex structure.

Main results:

New information technologies, prototype hardware, and software:

(1) information technology for efficient presentation, storage and transmission, access control, and protection of SSD in modern distributed information systems, including special hardware and software,
(2) instrumental software for processing, analyzing, and detecting observed objects under conditions of *a priori* uncertainty, including tools for simulation modeling and research into the efficiency of using the developed methods and algorithms for processing multidimensional signals,
(3) information technology and hardware and software tools for assessing a person's condition in extreme conditions and medical monitoring, based on automatic analysis of continuously recorded physiological data.

IV. Development of basic complexes of information technologies and software and hardware for processing and analyzing multidimensional and speech signals, including real-time systems.

The goal is the development, experimental research, software, and hardware-software implementation of complexes of methods, algorithms, and information technologies for processing one-dimensional and multidimensional signals (speech, images, video, etc.) in automatic systems, including real-time systems.

Main results:

A unified set of methods, algorithms, and new information technologies that use the specifics of advanced signal generation equipment (including scanning measurements), ensuring the quality and computational efficiency of processing, significantly exceeding the corresponding characteristics of known analogs:

(1) mathematical models of the formation of one-dimensional and multidimensional signals in automatic systems (including real-time systems), methods, and information technologies for efficient modeling and optimization of the information path of such systems,
(2) basic mathematical methods and algorithms for fast digital processing of multidimensional and speech signals: filtering

and reconstruction, data compression, feature extraction, classification, etc.,

(3) new information technologies for processing and analyzing multidimensional and speech signals for a number of real automatic systems that perform operational information analysis, in particular, real time.

V. Development of families of standardized knowledge bases and databases for the support, development, and implementation of critical technology at the federal level "Systems for Recognition and Synthesis of Speech, Text, and Images".

The goal is development, research, and software implementation:

(1) basic modules and systems that accumulate and use knowledge on image analysis and evaluation, providing adequate structural and functional capabilities to support "intellectualization" of the selection and synthesis of methods and algorithms for image processing and analysis,

(2) databases and knowledge bases about the properties of speech signals.

Subject of research:

(1) construction of models of strategies for selecting and applying processing methods used by specialists in solving IA and understanding problems,

(2) development, research, and implementation of methods for automating and combining image processing, analysis, and recognition algorithms,

(3) development, research, and implementation of knowledge representation methods for image analysis and understanding,

(4) construction of a hierarchical classification of image processing, analysis, and recognition algorithms,

(5) construction of a hierarchical classification of problems of image analysis and understanding, identifying basic complex and basic elementary problems,

(6) development of a uniform formal representation and methods for the decomposition of problems for problem classes and image classes,

(7) development of a uniform formal representation of image processing, analysis, and recognition algorithms,
(8) development of a glossary and thesaurus for the analysis and understanding of images,
(9) development of a hypertext help system on image analysis and evaluation,
(10) development and software implementation of systems for intellectualization of the selection and synthesis of methods and image processing and analysis algorithms,
(11) development and software implementation of libraries of descriptive algorithmic schemes for solving image processing and analysis problems for various types of transformations, problem classes, and image classes,
(12) development and software implementation of methods for automating the formation and processing of queries in the knowledge base,
(13) development of a class hierarchy for the system.

Main results:

(1) a prototype of a knowledge base that accumulates and uses knowledge on image analysis and evaluation in order to automate and increase the efficiency in solving image processing, analysis, and recognition problems,
(2) standard units of knowledge bases for image analysis and evaluation,
(3) composition of prototype of knowledge base:
    (a) intelligent support system,
    (b) algorithm database,
    (c) test image object database,
    (d) Lexicographic support database,
    (e) Information support database (hypertext, glossary),
    (f) software and tool environment,
(4) the architecture of the system provides representation of knowledge on image analysis and understanding, description of problems, planning, execution control, functioning of the knowledge base, and combination of algorithms,
(5) Windows OS,

(6) functional features of the system:

(a) combination of capabilities and functions of the system that accumulates and uses knowledge and an integrated software environment,
(b) standard dialog shell and developer tool environment,
(c) openness regarding objects processed by the system, algorithm libraries, and filling with lexicographic and expert information,
(d) the availability of a knowledge base on methods for analyzing and understanding images, constructed on the basis of a thesaurus of the subject area, containing tools to support semantic integration and ensuring the accumulation and use of knowledge,
(e) a uniform way of describing and representing problems, algorithms, and images,
(f) functional completeness of algorithmic content for problems within the system's area of expertise,
(g) availability of a set of databases that provide full support for all problems being solved,
(h) the ability to update the system by including new and excluding obsolete modules.

VI. Development of information, algorithmic, software, technical, and technological support for general-purpose intelligent geographic information systems.

The goals are theoretical and applied studies on the development and creation of intelligent general-purpose object-oriented topological geographic information systems.

Subject of research:

(1) models, methods, and tools for describing and representing initial spatially distributed information (SDI),
(2) models and methods of pattern recognition, image analysis, and structural description in problems of processing SDI,
(3) models and knowledge bases to support information technologies for processing PRIs,
(4) models and methods of decision-making in SDI processing problems,

(5) object-oriented databases to support geographic information technologies,
(6) methods of discrete geometry in problems of structural description and establishment of topological relations of SDI,
(7) multimedia approaches in information technology and SDI processing,
(8) computer graphics and display of SDI,
(9) information and technical support for geographic information systems,
(10) algorithmic and software and specialized hardware for geographic information systems and technologies.

Main results:

(1) a prototype of intelligent object-oriented general-purpose geographic information systems (GIS),
(2) a methodology for the development and experimental research of specialized intelligent GIS.

VII. Development of basic information technology systems and software and hardware for processing and analyzing remote sensing data for real-time assessment of the dynamics of events and phenomena on the Earth's surface and in the atmosphere.

The goal is to develop and create a set of information technologies and software and hardware tools for analyzing remote sensing data for real-time assessment of the dynamics of processes and phenomena on the Earth's surface and in the atmosphere:

(1) creation of a mathematical model of a remote sensing data perception system taking into account the influence of the atmosphere,
(2) development of methods and algorithms for joint processing of multispectral image sequences to detect and predict the dynamics of development of processes and phenomena,
(3) development of information technologies and software and hardware, including real time, to implement the created methods and algorithms,
(4) assessing the efficiency of developed methods and tools when solving applied problems.

Subject of research:

(1) development and research of a geometric model of observation systems and development of methods for estimating their parameters,
(2) development and research of statistical and information models of geospatial space monitoring data,
(3) development and research of methods for analyzing geospatial data, including methods for the following:
   (a) establishing correspondence when processing sequences of images and stereo images,
   (b) reconstruction of the surfaces of space scenes in space stereo vision,
   (c) joint processing of multispectral images to search for dynamic processes and phenomena on the Earth's surface and in the atmosphere,
   (d) prediction of the dynamics of their evolution,
   (e) identification of satellite images using digital database maps,
(4) architecture of global space vision systems: study of the possibilities of using channels of the worldwide telecommunication network and open system standards to create a space system for environmental and natural monitoring that operates in real and quasi-real time,
(5) architecture of local space vision systems: development of an information and software scheme for distributed processing of geospatial data based on local INTRANET protocols,
(6) high-resolution software and hardware for monitoring pollution of the Earth's surface and air basins using space observations,
(7) software and hardware for operational monitoring of natural disasters (forest fires, volcanic eruptions, etc.) using space observations,
(8) software and hardware for monitoring the condition of agricultural complexes using space observations.

Main results:

(1) methods and algorithms for analyzing and understanding real three-dimensional scenes during space monitoring of the Earth,

(2) hardware and software developments in the field of using computer vision methods in processing space images in real time,
(3) application packages for reconstructing the Earth's surface from space, detecting dynamic processes and phenomena, and predicting the dynamics of their development.

The main applied results of research and development in this scientific and technical field are promising information technologies — standardized unified systems of methods and tools providing efficient and high-precision solutions to applied problems of recognition, classification, and prediction based on high-tech algorithmic technologies and schemes based on new mathematical apparatus, efficient computational procedures, and data models that present source information in a form convenient for recognition. These technologies are designed for processing and analyzing data specified in the form of alphanumeric arrays, images, signals of various physical nature, texts, and their combinations and mixtures. Information technology is formalized and distributed in the form of algorithmic-software complexes and software tools.

3. The Main Scientific Schools of the RF in IA and PR at the Russian Academy of Sciences

This section lists the main scientific schools in IA and RE (and their leaders), created, developed, and operating on the basis of the foremost scientific institutes of the USSR Academy of Sciences, the Russian Academy of Sciences, and Russian universities:

1. Federal Research Center Computer Science and Control, Russian Academy of Sciences, Moscow, RF (Yu.I. Zhuravlev, V.L. Arlazarov, I.B. Gurevich, D.A. Pospelov, I.N. Sinitsyn).
2. Image Processing Systems Institute of the Russian Academy of Sciences, Branch of the Federal Scientific Research Centre Crystallography and Photonics, Russian Academy of Sciences, Samara, RF (V.A. Soifer).
3. Institute of Engineering Cybernetics — United Institute of Informatics Problems of the National Academy of Sciences of Belarus, Belarusian State University, Minsk, Belarus (C.V. Ablameiko).

4. Institute for Information Transmission Problems, RAS (Kharkevich Institute), Moscow, RF (D.S. Lebedev, I.A. Ovaseevich, V.L. Stefanyuk, L.P. Yaroslavsky).
5. Institute of Automation and Electrometry, Siberian Branch, RAS, Novosibirsk, RF (V.S. Kirichuk).
6. Institute of Mathematical Problems of Biology, RAS — Branch of the Keldysh Institute of Applied Mathematics, Russian Academy of Sciences, Pushchino-on-Oka, Moscow oblast, RF (Dedus, M.N. Ustinin).
7. Kotel'nikov Institute of Radio Engineering and Electronics, RAS, Moscow, RF (U.V. Obukhov).
8. N.N. Krasovskii Institute of Mathematics and Mechanics, Ural Branch, RAS, Yekaterinburg, RF (Vl.D. Mazurov).
9. Nizhny Novgorod State University, Nizhny Novgorod, RF (Yu.G. Vasin).
10. Moscow State University, Moscow, RF:

    10.1 Faculty of Computational Mathematics and Cybernetics (A.S. Krylov),
    10.2 Faculty of Physics (Yu.P. Pytyev),
    10.3 Skobeltsyn Institute of Nuclear Physics (I.G. Persiansev).

11. St. Petersburg Electrotechnical University LETI, St. Petersburg, RF (A.P. Nemirko).
12. Trapeznikov Institute of Control Sciences, RAS, Moscow, RF (M.A. Fizerman, V.N. Vapnyk, Y.Z. Tsypkin, A.Ya. Chervonenkis)
13. Ulyanovsk State Technical University, Ulyanovsk, RF (K.K. Vasiliev).
14. Yaroslav-the-Wise Novgorod State University, Veliky Novgorod, RF (H.M. Emelyanov).
15. Sobolev Institute of Mathematics, Siberian Branch, RAS, Novosibirsk, RF (N.G. Zagoruiko).
16. Mari El State Technical University, Yoshkar-Ola (Volga State University Of Technology), RF (Ya.A. Furman).
17. Research Design and Technology Institute of Biotechnical Systems, St. Petersburg State Electrotechnical University (under scientific and methodological supervision of the USSR Academy of Sciences/RAs), St. Petersburg, RF (V.M. Akhutin).

Note that the vast majority of scientific schools represented are still operating.

Detailed information about the cited scientific schools can be found in the Special Issue of the international scientific journal of the RAS Pattern Recognition and Image Analysis: Advances in Mathematical Theory and Applications "Scientific Schools of Scientists of the USSR, Republic of Belarus, and the Russian Federation in the Field of Image Analysis, Pattern Recognition and Related Problems of Computer Science" [1].

The main studies by scientists and specialists of the RF in the field of IA and PR since 1990 are currently published in English in the international peer-reviewed scientific journal, RAS, *Pattern Recognition and Image Analysis*: *Advances in Mathematical Theory and Applications* (Pleiades Publishing, Ltd., ISSN PRINT: 1054-6618; ISSN ONLINE: 1555–6212; see PRIA: http://pleiades.online and http://link.springer.com).

In the 1980s, thematic conferences began to be held in the USSR, where problems, methods, and applied problems of IA and PR were discussed. The materials of these conferences are also a good, reliable source for getting acquainted with the main results of Russian scientists and specialists in IA and PR.

The organizers of these conferences during 1986–2005 were the Computer Science section of the NSC RAS and, from 2005 to the present, the National Committee for Pattern Recognition and Image Analysis, RAS (member of the International Association for Pattern Recognition).

During 1986–2023, several cycles of such conferences have been held. A list of the most representative and scientifically significant conferences is given as follows:

1. Conferences "Pattern Recognition and Image Analysis: New Information Technologies":

   1st All-Union Conference "Pattern Recognition and Image Analysis: New Information Technologies" (ROAI-1-91), Minsk, October 14–18, 1991 (organizers: USSR Academy of Sciences; USSR State Committee for Science and Technology; Belarusian Regional Branch of the section "Computer Science" section of the NSC of the USSR Academy of Sciences; Academy of Sciences of the BelSSR; Belarusian State University).

2nd All-Russian conference with the participation of CIS countries "Pattern Recognition and Image Analysis: New information technologies" (ROAI-2-95), Ulyanovsk State Technical University, Ulyanovsk.

3rd All-Russian conference with the participation of CIS countries "Pattern Recognition and Image Analysis: New Information Technologies" (ROAI-3-97), Nizhny Novgorod State University, Nizhny Novgorod, December 1–6, 1997.

4th All-Russian conference with international participation "Pattern Recognition and Image Analysis: New Information Technologies" (ROAI-4-98), Institute of Automation and Electrometry, SB RAS, Novosibirsk, October 11–18, 1998.

5th International Conference "Pattern Recognition and Image Analysis: New Information Technologies" (ROAI-5-2000), Institute of Image Processing Systems (IIPS RAS), SSAU, Samara, October 16–22, 2000.

6th International Conference "Pattern Recognition and Image Analysis: New Information Technologies" (ROAI-6-2002), NovSU, Veliky Novgorod, October 21–26, 2002.

7th International Conference "Pattern Recognition and Image Analysis: New Information Technologies" (ROAI-7-2004), St. Petersburg State Electrotechnical University (LETI), St. Petersburg, October 18–23, 2004.

8th International Conference "Pattern Recognition and Image Analysis: New Information Technologies" (ROAI-8-2007), Mari El State Technical University, Yoshkar-Ola, October 7–13, 2007.

9th International Conference "Pattern Recognition and Image Analysis: New Information Technologies" (ROAI-9-2008), Nizhny Novgorod State University, Nizhny Novgorod, September 14–20, 2008.

10th International Conference "Pattern Recognition and Image Analysis: New Information Technologies" (ROAI-10-2010), St. Petersburg State Electrotechnical University (LETI), December 5–12, 2010.

11th International Conference "Pattern Recognition and Image Analysis: New Information Technologies" (ROAI-11-2013), IISP RAS, SSAU, Samara, September 23–28, 2013.

2. Open German–Russian Workshop on Pattern Recognition and Image Understanding:

   1st Open German–Russian Workshop on Pattern Recognition and Image Understanding (OGRW-1-1990), Berlin, Germany, 1990.

   2nd Open Russian–German Workshop on Pattern Recognition and Image Understanding (OGRW-2-1991), St. Petersburg, RF, 1991.

   3rd Open German–Russian Workshop on Pattern Recognition and Image Understanding (OGRW-3-1993), Erlangen, Germany, 1993.

   4th Open Russian–German Workshop on Pattern Recognition and Image Understanding (OGRW-4-1996), Valdai, RF, March 3–9 1996.

   5th Open German–Russian Workshop on Pattern Recognition and Image Understanding (OGRW-5-1998), Herrsching, Germany, September 21–25, 1998.

   6th Open Russian–German Workshop on Pattern Recognition and Image Understanding (OGRW-6-2003), Katun Village, Altai krai, RF, August 25–30, 2003.

   7th Open German–Russian Workshop on Pattern Recognition and Image Understanding (OGRW-7-2007), Ettlingen, Germany, August 20–23, 2007.

   8th Open Russian–German Workshop on Pattern Recognition and Image Understanding (OGRW-8-2011), Nizhny Novgorod, RF, November 21–26, 2011.

   9th Open German–Russian Workshop on Pattern Recognition and Image Understanding (OGRW-9-2014), Koblenz, Germany, December 1–5, 2014.

3. IAPR TC 16 International Workshop “Image Mining: Theory and Application”:

   IMTA-I-2008, Funchal, Madeira, Portugal, in conjunction with the 3rd International Conference on Computer Vision Theory and Applications (VISAPP 2008).

   IMTA-II-2009, Lisboa, Portugal, in conjunction with the 4th International Joint Conference on Computer Vision, Imaging, and Computer Graphics Theory and Applications (VISIGRAPP 2009).

IMTA-III-2010, Angers, France, in conjunction with the 5th International Joint Conference on Computer Vision, Imaging, and Computer Graphics Theory and Applications (VISIGRAPP 2010).

IMTA-IV-2013, Barcelona, Spain, in conjunction with the 8th International Joint Conference on Computer Vision, Imaging, and Computer Graphics Theory and Applications (VISIGRAPP 2013).

IMTA-V-2015, Berlin, Germany, in conjunction with the 10th International Joint Conference on Computer Vision, Imaging, and Computer Graphics Theory and Applications (VISIGRAPP 2015).

IMTA-VI-2018, Montreal, Canada, in conjunction with the 1st International Conference on Pattern Recognition and Artificial Intelligence (ICPRAI 2018).

IMTA-VII-2021, Milano, Italy, in conjunction with the 25th International Conference on Pattern Recognition (ICPR 2020).

IMTA-VIII-2022, Montreal, Canada, Italy, in conjunction with the 26th International Conference on Pattern Recognition (ICPR 2022).

4. Conferences "Automated Image Processing Systems":

   1st conference "Automated Image Processing Systems" (ASO-Iz-81), Moscow, 1981.

   2nd All-Union Conference "Automated Image Processing Systems" (ASO-Iz-86), Lvov, September 1986.

   3rd All-Union Conference "Automated Image Processing Systems" (ASO-Iz-89), Leningrad, 1989.

4. Conclusions

The collective volume contains an introductory chapter by the co-editors, Igor B. Gurevich and Vera V. Yashina, and 27 chapters, which are divided into 5 parts as follows:

Part I. Mathematical Theory of Image Analysis
Part II. Mathematical Theory of Pattern Recognition
Part III. Applications of Image Analysis and Pattern Recognition

Part IV. Algebraic Approach to Classification, Forecasting, and Pattern Recognition Problems
Part V. Human Vision And Speech Analysis.

The invited authors comprise leading scientists and specialists of the RAS and foremost Russian universities representing the main scientific schools in the field of IA and PR in the RF. The monograph is a representative slice of the state of the art of fundamental and applied research in IA and PR in the RF.

Analysis of the materials shows that the main goals of research in IA and PR are as follows:

(1) development of fundamental mathematical foundations for promising methods of machine data analysis (mathematical and numerical models, methods, and algorithms),
(2) creation of new promising technologies for computer data analysis (bringing developed models, methods, and algorithms to the level of applied technologies and prototype systems),
(3) development of new-generation (based on constructive computational AI methods) technologies aimed at creating reliable environments for intelligent analysis of Big Data (including in cloud computing environments),
(4) development of application systems (operating algorithmic-software prototype systems) oriented toward application problems in dual-use and security technologies, high-tech medicine and molecular pharmacology, monitoring and analysis of financial flows, and computer processing of large arrays of full-text documents,
(5) development of new methods for detecting empirical patterns in sequences of expanding fact bases and fast similarity detection algorithms,
(6) development of efficient methods for analyzing and classifying partially ordered data,
(7) development of methods and algorithms for computer algebra, including in application to solving the cited problems,
(8) development of numerical prediction methods (including in the field of prediction of the properties of new-generation composite materials for high-tech areas of the economy),
(9) development of methods for processing experimental data to correct numerical simulation results,

(10) development and research of descriptive methods and models for image analysis and evaluation based on the joint use of multialgorithmic classifiers and dual representations of source data,
(11) interpretation of developed descriptive image analysis methods in biomedical image analysis.

It is also becoming obvious that development of the mathematical theory of IA follows the path of its algebraization based on the algebraic approach to recognition and classification problems, and its specialization in the case of representing initial information in the form of images: descriptive theory of image analysis, as well as development of the theory of image algebras.

Analysis of the main directions of research of scientific schools of the RF in IA and PR and their most important results leads to the conclusion that the problem of analysis and evaluation of difficultly formalized, complexly structured, incomplete, contradictory, inhomogeneous information is extremely relevant and promising both in the development of fundamental research and the use of new computationally efficient mathematical methods in solving the entire diversity of applied problems.

Thus, the state of the art of this area of computer science is characterized by the following provisions:

(1) the fundamental research being carried out generally corresponds to the world level, and in some approaches, leads it (e.g. descriptive image analysis),
(2) promising areas of development:
    (a) new formulations of mathematical problems arising in connection with modeling, generation, and construction of representations of algorithmic schemes and IA and PR objects,
    (b) expansion and combination of tools from various branches of mathematics used in the development of new mathematical IA and PR methods,
    (c) mathematical methods for automating extraction of knowledge and information from images,
    (d) the need to reduce the gap between the potential capabilities of the mathematical theories of IA and PR and their use in solving applied problems,

(e) the need for objective analysis, based on mathematical research, of the real capabilities, accuracy, and interpretability of the results of neural network methods in IA and PR problems,

(3) construction of a general theory of image analysis is far from complete,

(4) wide commercialization of the results of mathematical theories of IA and PR has not led to a corresponding expansion of the scope and development of fundamental research.

In conclusion, the scientific coeditors of the collective monograph are grateful to **Professor C. Suen, editor of the books Series on Language Processing, Pattern Recognition, and Intelligent Systems** for the invitation to publish a book in this series and to the authors for submitting high-quality original scientific articles for publication.

This book is dedicated to **the 300th anniversary of the Russian Academy of Sciences.**

Reference

[1] Special Issue "Scientific Schools of Scientists of the USSR, the Republic of Belarus and the RF in the Field of Image Analysis, Pattern Recognition and Related Problems of Computer Science," Ablameyko, S.V., Gurevich, I.B., and Yashina, V.V. (eds.) (2023) *Pattern Recognition and Image Analysis: Advances in Mathematical Theory and Applications*, 33(4).

© 2025 World Scientific Publishing Company
https://doi.org/10.1142/9789811267215_fmatter

Contents

About the Editors v

Image Analysis and Pattern Recognition in the Russian Federation: Main Results and Development Trends ix
Igor B. Gurevich and Vera V. Yashina

Part I. Mathematical Theory of Image Analysis

1. Descriptive Image Analysis: Descriptive Image Algebras **1**
Igor B. Gurevich and Vera V. Yashina

2. Morphological Image Analysis: Mathematical Foundations and Applications **81**
Yuri P. Pyt'ev, Alexey I. Chulichkov, Olesya V. Falomkina, Dmitriy A. Balakin, and Andrey V. Zubyuk

3. Image Analysis and Processing Based on Two-Dimensional Variations **119**
Pavel A. Chochia and Olga P. Milukova

4. Representation and Analysis of Regular Textures in Radon Space **179**
Ivan G. Kazantsev

5. Mathematical Model and Methods of Analysis of Continuous Contours Images **199**

Rinat G. Khafizov, Sergei A. Okhotnikov, and Mikhail A. Egoshin

6. Modeling and Processing of Image Sequences **233**

Konstantin K. Vasiliev, Viktor R. Krasheninnikov, and Vitaliy E. Dementev

7. Conjugation Index as a Proximity Measure in Problems of Recognition **293**

Vladimir A. Fursov, Sergey A. Bibikov, and Denis A. Zherdev

8. Target Shape Identification on Low-Count Images Using Expectation–Maximization Algorithms **323**

Vyacheslav E. Antsiperov and Aleksandr S. Bugaev

9. Nonlinear Mapping of Hyperspectral Data **363**

Evgeny V. Myasnikov

10. Intellectual Software Support in Problems of Digital Registration of Random Point Structures **391**

Alexandr L. Reznik and Alexandr A. Soloviev

Part II. Mathematical Theory of Pattern Recognition

11. Collective Generalized Solutions for Infeasible Systems of Constraints and Ensemble Learning Techniques **459**

Vladimir D. Mazurov and Mikhail Yu. Khachay

12. Some Collective Methods in Cluster Analysis and Regressions **489**

Vladimir V. Ryazanov

13. Solving High-Dimensional Inverse Problems with Artificial Neural Networks **505**

Sergey A. Dolenko, Igor V. Isaev, Alexander O. Efitorov, Sergey A. Burikov, Tatiana A. Dolenko, Eugeny A. Obornev, and Mikhail I. Shimelevich

14. Generalized Spectral-Analytical Method and Its Applications in Problems of Image Analysis and Pattern Recognition **545**

Sergey A. Makhortykh, Liudmila I. Kulikova, Anton N. Pankratov, and Ruslan K. Tetuev

Part III. Applications of Image Analysis and Pattern Recognition

III.1 Biomedical Applications

15. Recognition of Biomedical Signals for Medical Diagnostics **579**

Anatoly P. Nemirko and Ludmila A. Manilo

16. Brain Signal Analysis from the Standpoint of the Structural Approach **615**

Yakov A. Furman, Victor V. Sevastyanov, and Konstantin O. Ivanov

17. The Early Stages of Post-traumatic Epileptiform Activity and Parkinson's Disease Based on EEG Wavelet Spectra Features **657**

Yury V. Obukhov, Konstantin Yu. Obukhov, Ivan A. Kershner, and Ilya G. Komoltsev

18. Search of Completely Verified Regularities in Biomedical Data **677**

Oleg V. Senko and Anna V. Kuznetsova

19. Mathematical Methods of the Blood Vessels Analysis on the Biomedical Images **699**

Natalya Yu. Ilyasova, Nikita S. Demin, and Rustam A. Paringer

20. Joint Flow Unwrapping and 2D Flow Restoration in Cardiac Color Doppler Flow Mapping **729**

Artem M. Yatchenko, Andrey S. Krylov, Valery A. Sandrikov, and Tatiana Yu. Kulagina

21. Computer Analysis of Single Point Radiator Fluorescent Images **745**

Aleksey A. Rozhentsov and Aleksey A. Baev

III.2 GIS Technologies

22. Models, Methods, and Information Technologies for Creating, Processing, and Analyzing Spatially Distributed Data **771**

Yury G. Vasin, Leonid I. Lebedev, and Tamara Sh. Utesheva

23. Methods for Aerospace Images and Raster Topographic Maps Processing to Create and Update Geospatial Data **799**

Vitaly S. Titov, Sergey Yu. Miroshnichenko, and Vladimir S. Panishchev

III.3 Document Analysis

24. Mobile ID Recognition: Coarse-to-Fine Approach **827**

Vladimir L. Arlazarov, Vladimir V. Arlazarov, Konstantin B. Bulatov, Timofey S. Chernov, Dmitry P. Nikolaev, Alexander V. Sheshkus, Natalya S. Skoryukina, Oleg A. Slavin, and Sergey A. Usilin

Part IV. Algebraic Approach to Classification, Forecasting and Pattern Recognition Problems

25. Multilayer Constructions of the Algebraic Approach in Pattern Recognition **915**

Yury I. Zhuravlev, *Konstantin V. Rudakov*, *and Yury G. Smetanin*

Part V. Human Vision and Speech Analysis

26. Multilevel Models of Active Vision in Image Analysis **937**

Mikhail V. Petrushan, Mikhail V. Kopeliovich, Konstantin B. Kalinin, Boris M. Vladimirskiy, Dmitry G. Shaposhnikov, and Anatoly I. Samarin

27. Implicit Modeling of Pronunciation Variation in Speech Recognition Systems **975**

Vladimir Ya. Chuchupal

Index 995

This page intentionally left blank

© 2025 World Scientific Publishing Company
https://doi.org/10.1142/9789811267215_0001

Chapter 1

Descriptive Image Analysis: Descriptive Image Algebras

Igor B. Gurevich* **and Vera V. Yashina†**

Federal Research Center "Computer Science and Control" of the Russian Academy of Sciences, 44/2, Vavilov str., Moscow, the Russian Federation
**igourevi@ccas.ru*
†*werayashina@gmail.com*

This chapter deals with a new class of specialized algebras, viz. descriptive image algebras, which is an original mathematical language used to formalize and standardize the processing procedures for image models and transformations over them. Descriptive image algebras are used to describe problems, objects, and transformations involved in extracting information from images. When speaking of descriptive image algebras in this publication, we consider descriptive image algebras with one ring in more detail. This class of algebra falls into the class of universal linear algebra with sigma-associative ring with identity. Classes of descriptive image algebras with two or more rings fall into the class of homogeneous graded algebras. Descriptive image algebras are the principal branch of the mathematical apparatus of descriptive image analysis, which is a logically structured totality of descriptive methods and models designed for image analysis and estimation. Up to now, the following principal results associated with "algebraization" have been obtained within the framework of descriptive image analysis: (1) descriptive image algebras are introduced and defined as a mathematical object, (2) the class of descriptive image algebras with one ring is introduced and studied, (3) descriptive image algebras with one ring are defined with the method and necessary and sufficient conditions for constructing them proposed, (4) the specialized versions of descriptive image algebras with

https://avxhm.se/blogs/hill0

one ring over images, models, and image representations as well as over transformations of image models and images are defined, (5) the set of operations of the standard image algebra that ensure construction of descriptive image algebras with one ring is defined, (6) classes of descriptive image algebras with one ring that generate the classes of image models are defined, and (7) the mathematical and functional/physical interpretation of image analysis and processing operations used as the sets of operations (the elements of the ring) of descriptive image algebras with one ring is studied. The following results of the studies obtained within the descriptive image analysis are studied, viz. (1) image models (representations and formalized descriptions) and (2) the algebraic model for solving the problem of automated ophthalmological diagnostics. The principal publications that deal with image algebras are listed in references.

Keywords: Image analysis, descriptive image analysis, mathematical theory of image analysis, descriptive algorithmic schemes, descriptive image models, image representations, representations of image analysis procedures, descriptive approach to image analysis, image algebras, image models, biomedical images.

1.1 Introduction

This chapter deals with a new class of specialized algebras, viz. descriptive image algebras (DIAs). DIAs are used to describe problems, objects, and transformations considered in the course of information extraction from images. When we describe algorithmic schemes of image formal representation, processing, analysis, and recognition using DIA, each element of the scheme and any transformation used in the scheme are given by the structures constructed by applying DIA operations to the set of DIA operands. With such approaches, we can vary methods of solving the subproblem by using image analysis operations as DIA elements while preserving the overall scheme of the technique of information extraction from images.

When speaking of DIA in this publication, we consider DIA with one ring (DIA1R) in more detail. This class of algebra falls into the class of universal linear algebra with sigma-associative ring with identity. Figure 1.1 shows the position of DIA1R in the general classification of algebra types [40]. DIA classes with two or more rings fall into the class of homogeneous-graded algebras (see Figure 1.1).

DIA is the principal branch of the mathematical apparatus of descriptive analysis (DA) of images [47–56], which is a logically

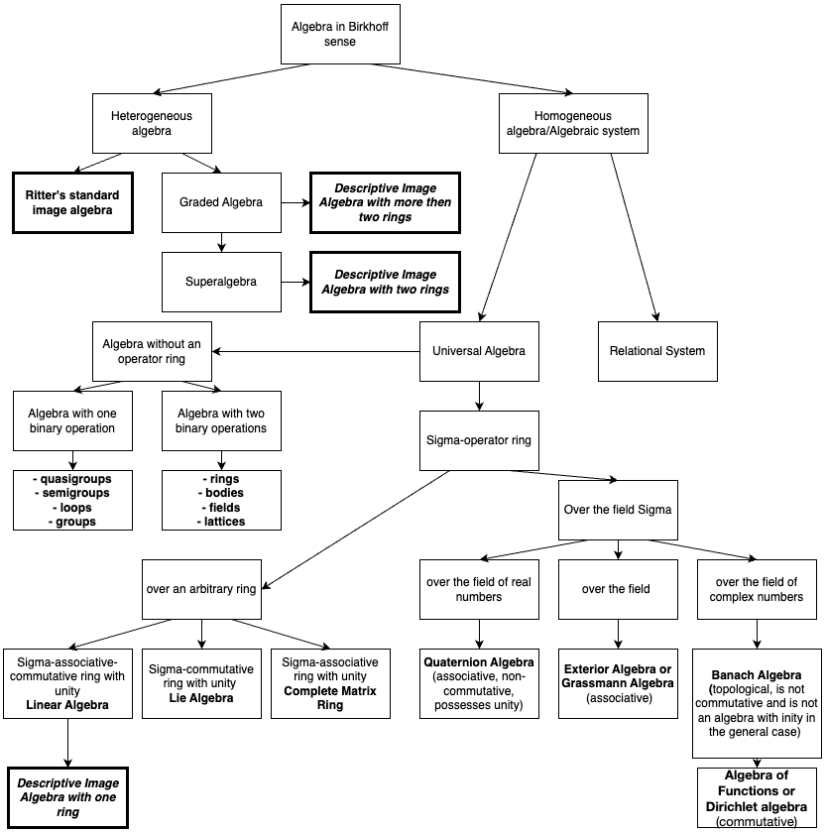


Figure 1.1. Algebra classification.

arranged totality of descriptive methods and models designed for image analysis and estimation.

In applied mathematics and computer science, construction and application of mathematical and imitational models of objects and procedures used to transform them are the conventional standardization methods.

Modern state and trends of DA development are specified by methods, models, and results of the descriptive approach (DAp) to image analysis and understanding proposed by Gurevich and developed by his fellow scholars and students [27,29]. Since 2002, Yashina has made a significant contribution to the theory. As its methods and apparatus were being developed and refined, the interpretation of the descriptive approach was proposed and defined as DA.

https://avxhm.se/blogs/hill0

The mathematical basis of image analysis and recognition is now studied [47–56] in the Federal Research Center "Computer Science and Control" of the Russian Academy of Sciences.

In the course of developing and implementing DAp [27–46], a new class of image algebras — DIA — is introduced, specified, and studied [40, 49], the principal types of image models (descriptive image models, DIMs) are introduced, classified, and studied [45,51], axioms of the descriptive theory of image analysis are introduced [46], the general model of the image recognition process is defined and studied [29], new image analysis and recognition problem statements are introduced [31,44], the notion of equivalence of images in the recognition problem is introduced and studied [44], new spatial classes of image recognition algorithms are defined and studied [37], the notion of generating descriptive trees (GDTs) is introduced [42], and the image formalization space (IFS) is defined and studied [46].

In what follows, all methods and results we give here relate to DA. The principal DA goal [47–56] is to arrange and standardize various methods, processes, and concepts used in image analysis and recognition.

DA solves fundamental problems of formalization and systematization of methods and forms of information representation in image analysis, recognition, and understanding problems. In particular, it solves problems associated with automatization of information extraction from images in order to make intellectual decisions (diagnostics, forecasting, detection, evaluation, and identification of patterns).

Automatization of information extraction from images requires comprehensive use of all functionality of the mathematical apparatus in image analysis, recognition, and understanding problems. DA axiomatics and formal structures provide the methods and tools for representation and description of images for them to be analyzed and evaluated.

DAp, general algebraic methods, methods of mathematical theories of image processing, image analysis, and pattern recognition form the theoretical basis for the studies.

DA allows solving both problems associated with construction of formal descriptions of images as recognition objects and problems of synthesizing image recognition and understanding procedures. It is also assumed that the processes of analysis and evaluation of data represented as images (trajectories of solving the problem) could be

basically treated as a sequence/combination of transformations and computations of the totality of intermediate and final (determining the solution) estimates. These transformations are specified on the classes of equivalence of images and their representations. The latter are given descriptively, i.e. using a basic set of prototypes and respective generating transformations functionally complete with respect to the class of equivalence of admissible transformations.

DA includes the following principal tools of mathematical theory of pattern recognition and mathematical theory of image analysis:

(1) the totality of pattern analysis and recognition methods,
(2) methods of reducing images to the convenient-for-recognition form (RCRF) [29, 45, 46, 53],
(3) conceptual system of pattern analysis and recognition [29, 45, 46],
(4) DIM [29, 45, 46],
(5) the universal DIA language [40, 44, 49],
(6) mathematical statements of pattern analysis and recognition problems [29, 31, 55],
(7) basic pattern recognition models [55, 56],
(8) models of the process of solving image recognition problems [53, 54, 56],
(9) descriptive algorithmic schemes (DAS) [53, 54, 56], including DAS of image representation (DASIR) [41, 46],
(10) multimodel and multiaspect image representations based on GDT [42].

The principal directions of research within DA are

(1) creating axiomatics of image analysis and recognition;
(2) developing and applying the single language for describing image analysis and recognition processes (studying DIA);
(3) introducing formal systems based on regular informational structures to specify image analysis and recognition processes.

The fundamental methodological DA principles imply the following:

(1) "algebraization" of image analysis, in particular using the standard algebraic DIA language to describe
    - models and representations of images and procedures used to construct and transform them,

https://avxhm.se/blogs/hill0

- procedures of transformation over models and representations that ensure the stated image analysis and evaluation problems can be solved,

(2) standardization of representation of image analysis and recognition problems,
(3) conceptualization and formalization of representations (phases) passed through by an image while it is being transformed in the course of solving the recognition problem,
(4) classification and specification of admissible image models (AIM),
(5) RCRF,
(6) combination of algorithms in multilevel multialgorithmic models of image classifiers,
(7) using multimodel and multiaspect image representations,
(8) constructing and using basic models of the process of solving the standard pattern recognition problem,
(9) using the non-classical mathematical recognition theory to specify new statements of image analysis and recognition problems.

The following principal theories of pattern recognition and image analysis have made a significant impact on creating DA, viz.

(1) the algebraic approach to image recognition by Zhuravlev [139] and his algebra of algorithms,
(2) Grenander's pattern theory [23], in particular, the algebraic methods for representation of original data in recognition problems,
(3) Marr's informational vision theory [79].

As we noted above, within DA, we propose to perform "algebraization" of image analysis and recognition using DIA [40, 44, 49]. This new class of image algebras was developed based on research in the domain of "algebraization" of pattern recognition and image analysis performed since the 1970s. Creation of the new algebra was directly influenced by the works of Sternberg [125] and Ritter [102], who defined the classical variants of image algebras.

For a more detailed description of DA methods and tools obtained while developing its results, see [27–56].

Up to now, the following principal results have been obtained within DA:

(1) "Algebraization" of image analysis, viz.

- DAp characterization is performed [29],
- DAp axioms are introduced and justified [46],
- DIA is introduced and defined as a mathematical object [40],
- new statements of image recognition problems are introduced and justified [29, 31, 44, 53, 54],
- DIA1R are introduced and studied [40],
- for DIA1R, the definition is given and the construction method and necessary and sufficient conditions are proposed [40, 43],
- the specialized versions of DIA1R over images, image models and representations, and transformations of image models and images are specified [43, 49, 50, 52],
- the set of operations of the standard image algebra that ensure DIA1R construction is specified [43],
- DIA1R classes that generate classes of image models are specified [43],
- the mathematical and functional/physical interpretation of image analysis and processing operations used as sets of operations (elements of the ring) of DIA1R is studied [49, 50, 52].

(2) Efficient ways and tools of image description and representation in recognition problems, viz.

- IFS is introduced, defined, and studied [46],
- IFS structure is specified [46],
- IFS topological properties are studied [46],
- DASIR is introduced, defined, and studied as a mathematical object [46],
- the DASIR classification is constructed [46],
- the standardization of DASIR representation is performed and the examples of DASIR for solving applied problems of biomedical image analysis are constructed [46],
- the informational structure for generating descriptive algorithmic image recognition schemes (DAIRS) is introduced [54],
- image representation is introduced and specified as a mathematical object [46],
- types of image representations are introduced and specified [46],
- DIM is introduced and specified as a mathematical object [45, 51],

https://avxhm.se/blogs/hill0

- DIM types are introduced, specified, and studied [45, 51],
- the DIM-based model of solving image recognition problems is proposed and implemented [29, 55, 56],
- the method for choosing the image transformation that allows taking into account the informational nature of images is proposed and its software implementation is developed [34, 35],
- the criteria for classification of features used for image description are stated [33],
- the classification of image features as a tool for formal description of image representations is constructed [33, 88],
- the method for construction of multiaspect image descriptions based on classifications of image features is proposed [42].

(3) Efficient algorithms for analysis, recognition, and evaluation of two-dimensional spatial information, viz.

- the multilevel model of combining the algorithms and original data in image recognition, which is based on joint usage of multialgorithmic classifiers and dual representations of original data of the "combinatorial structures of local neighborhoods/formalized models" type, is introduced and justified [53],
- the class of efficient recognition algorithms based on calculating estimates that allow for spatial data (DAEC) is specified and studied [26],
- the DAEC class with quadratic support sets is specified and studied [26],
- the method of efficient implementation of DAEC algorithms with systems of rectangular support sets and their compositions that is based on two-step procedures of search for rectangles is proposed [36],
- the methods of efficient implementation of DAEC algorithms with systems of rectangular support sets and proximity functions on pairs of support sets are proposed [37],
- the method for constructing and estimating the computational complexity of DAEC algorithms with systems of support sets generated by arbitrary references based on multistep procedures of searching for references on raster is proposed [37].

(4) Linguistic and knowledge-oriented tools for support of image analysis automatization, viz.

- the concept, architecture, and functional scheme of the referral Internet resource on image processing, analysis, and recognition are proposed and implemented [4],
- classification of image processing, analysis, and recognition problems is constructed [39],
- Automated System of Knowledge Base on Image Processing, Analysis, and Recognition is developed [3],
- Information and Search Thesaurus for Image Analysis is developed [3],
- Thesaurus of Cytohaematological Terms for Automatization System for Image Analysis of Blood Cells is developed [3, 4],
- the experimental version of image analysis ontology in OWL is developed [5, 9, 10, 38],
- Knowledge Base on Image Processing, Analysis, and Recognition is developed [3, 4].

(5) Developments, viz.

- Open System for Automating of Image Processing, Analysis, and Recognition [32],
- the Black Square library of calculating image features for the software and tool package of automatization of scientific research and learning in image processing, analysis, and recognition [32],
- Automated System for Analysis of Biomedical Micro Images for Detection and Characterization of Informative Objects of the Given Type against Inhomogeneous Background [57],
- Automated System for Analysis of Ophthalmologic Images of Various Classes [30, 57].

This chapter consists of introduction, four principal sections, conclusions, and references.

Section 1.2, Descriptive Image Models, deals with constructing image models (representations and formalized descriptions); in particular, according to three classes of admissible image transformations, we construct examples of three DIM classes, viz. procedural, parametric, and generating ones.

Section 1.3, Descriptive Image Algebras, defines the principal specifics of the DIA mathematical object, gives necessary definitions and explanations needed to introduce the original class of image algebras, and describes the results of studying the mathematical and

https://avxhm.se/blogs/hill0

functional/physical interpretation of image analysis and processing operations used as sets of DIA operations.

Section 1.4, Classes of Descriptive Image Algebras with One Ring, deals with constructing 15 classes of specialized DIA1R over the following operands — (1) image implementations, (2) image transformations, (3) image representations, and (4) image models — and over the following operations, viz. (1) standard algebraic operations, (2) transformations of image analysis and processing, and (3) algebraic closures, linear combinations, and superpositions of these operations.

Section 1.5, Algebraic Model of Solving the Problem of Automatization of Detection of Abnormal Changes in Morphometrical Characteristics of Human's Fundus, gives the algebraic model of solving the problem of automatization of ophthalmological diagnostics, viz. analysis of angiogram images obtained by the method of optical coherence tomography (OCT-A) used to automate detection of abnormal changes in morphometrical characteristics of human's fundus.

References include 139 publications, including all our principal works on DIA.

We give the list of abbreviations used in this chapter:

- DA — descriptive image analysis,
- DAEC — the class of efficient recognition algorithms based on the calculation of estimates that allow for spatial data,
- DIA — descriptive image algebras,
- DIA1R — descriptive image algebras with one ring,
- DAS — descriptive algorithmic schemes,
- DASIR — descriptive algorithmic schemes of image representation,
- DIM — descriptive image models,
- Dap — descriptive approach to image analysis and understanding,
- DAIRS — the informational structure for generating descriptive algorithmic image recognition schemes,
- CSLN representations — image representations of the "combinatorial structures of local neighborhoods" type,
- MAC — multialgorithmic classifiers,
- MMCAO — the multilevel model of combining algorithms and original data in image recognition based on joint usage of multialgorithmic classifiers and dual representations of original data of the "combinatorial structures of local neighborhoods/formalized models" type,

- OCT — optical coherence tomography,
- OCT-A — optical coherence tomography angiography,
- RCRF — reducing images to the convenient-for-recognition form,
- GDT — generating descriptive trees,
- IFS — image formalization space,
- FAA — foveolar avascular area.

1.2 Descriptive Image Models

1.2.1 *Reducing images to the convenient-for-recognition form*

Problems of intellectual decision-making based on data analysis require using some formal representation of original information — ideally, a mathematical model.

The principal DA idea [47,51,53,54] is to reduce all image analysis (processing, recognition, and understanding) processes to perform the following:

(1) constructing image models (representations, formalized descriptions),
(2) specifying transformations over image models,
(3) constructing models (representations and formalized descriptions) of transformations over image models and representations,
(4) constructing models (representations and formalized descriptions) of schemes of transformations over image models and representations that ensure solving image analysis problems.

In Section 1.2, we consider models of the first type — models of recognition objects, viz. images — and define transformations over image models.

We consider in more detail original data and their representation in the process of solving the image analysis and recognition problem.

Due to the extremely complex informational nature of an image and the technical peculiarities of forming and implementing digital image representation, one fails to construct a classical mathematical model of an image as an information object.

When working with images, the correct result can be achieved if the analysis relies on using some formalized image description that

https://avxhm.se/blogs/hill0

properly represents it, maximally corresponds to *a priori* knowledge and context and is compatible with the problem to be solved. In this sense, synthesizing a formalized image description is an optimization problem such that its solution should be stable, satisfy time and other resource restrictions given by the nature of the applied problem, and, where possible, possess certain generality, i.e. be applicable to new problems.

The original information represented as images is weakly structured and non-formalized and generally prevents standard recognition algorithms from being directly applied to it.

Intrinsically, an image is an object with a complex information structure that reproduces information about the original scene via the values of brightness of the discrete elements of the image, viz. pixels, configurations of image fragments, sets of pixels, and spatial and logical relations between configurations, sets of pixels, and particular pixels. Images differ from other means of data representation since they are highly informative, illustrative, structured, and natural in perception of its content for a human.

An image is a mixture of original (raw and "real") data, their implementations, and some deformations. Implementations (like the respective descriptions) reflect informational and physical nature of objects, events, and processes reproduced by an image while deformations are generated by technical characteristics of the tools used to register, form, and transform the image in the course of constructing its digital representations and hierarchy of their descriptions.

When we work with such types of information as symbolic and numerical information (results of observation, measurements, examination, numerical characteristics, parameters, expert opinions, etc.), formal descriptions of recognition objects — models — are comparatively simple while requirements on how to arrange and represent the original information are minimal as it is quite conceivable to restrict oneself to simple sets or tables of characteristics of the "feature–feature value" type. The principal peculiarity of such models of recognition objects is that they are represented by some set of characteristics, with their interconnection and relationships generally speaking ignored. Each object is also assumed to be identified with some point of a multidimensional feature space, while a class of objects is represented by a "compact" set of such points.

In the course of recognition (i.e. when an image is formalized as a recognition object), the "content" of the image is characterized by information of the following types, viz. (a) identifiable objects with well-defined structure, (b) identifiable objects with ill-defined structure, (c) non-identifiable objects, (d) identifiable interest zones with well-defined structure, (e) identifiable interest zones with ill-defined structure, (f) and all other, including the background that has nothing to do with information from (a)–(e).

There exist two principally different types of image representations, viz.

(1) to use the system of objects and relations singled out on an image [45],
(2) to use representations of the "combinatorial structures of local neighborhoods" type (CSLN representations) — spatial information itself (for instance, image fragments, which are pixels or totalities of pixels that form a local neighborhood of some order on an image).

Semantic information represented on an image or associated with the scene to be reproduced allows us to use, in the course of analysis and recognition, various context information (prohibited and impossible order relations, partial order relations, as well as other conditions and restrictions that reflect physical and logical structure of the real world). To apply a recognition algorithm, we need to use RCRF procedures.

In the first case, RCRF procedures should ensure construction of the mathematical model of an image. In DA, we choose DIM, viz. parametric, procedural, and generating models, as formalized models [45].

In the second case, the recognition algorithm should allow for original data that are basically an image or its fragments. Recognition is performed using "spatial" algorithms (algorithms that allow for spatial information as the original one) [26, 36, 37], and, in the general case, RCRF procedures can be reduced to image segmentation, choice of the form of its fragments used to compare the object to be recognized with the reference object, decomposition of the image into fragments, etc.

https://avxhm.se/blogs/hill0

1.2.2 *Image models and representations*

When defining the system of DA notions, we took the following image properties as the fundamental ones [46]:

(1) an image consists of a set of points and a set of values matched with these points;
(2) an image transformation aimed at their analysis includes operations over images and over various values and characteristics matched with these images;
(3) an image is characterized by two types of information, viz. it is given by spatial relations between points and by some types of numerical or some other descriptive information matched with these points;
(4) a set of points is a topological space (it consists of objects called points and a topology that specifies the proximity of two points, the connectivity of a subset of the set of points, the neighborliness of points, and the presence of boundary points, curves, and arcs).

Definition 1.1 ([46]). In recognition problems, the original **image** $\Im$ is a somehow ordered set of registered initial spatial and context data that reflects the type (form and state) of objects, events, and processes of the depicted scene and allows RCRF transformations to be applied to it.

Since each scene can be matched to some totality of images that differ by scale, observation angle, illumination, technical characteristics of the system of image registration, etc., we introduce the notion of image implementation.

Definition 1.2 ([46]). The **implementation** I' of the image $\Im$ is the set $I' = \{(x, f(x))\}_{x \in D_f}$ of points x that belong to the definition domain D_f of the image implementation I' and the set of values $f(x) \in F$ at each point of D_f. The function $f: D_f \to F$, F is any given set of values.

We introduce the following designations, viz. $\{I_0\}$ is the set of all possible original information on the image, $\{I'\}$ is the set of all digital implementations of the image, and $\{B\}$ is the set of additional information on the image (the set of context and semantic information associated with the ways to obtain and form the image and/or

reflecting some of its specific aspects associated with the nature of the depicted scene).

Lemma 1.1 ([45]). *The set of original pieces of information* $\{I_0\}$ *consists of two sets* $\{I'\}$ *and* $\{B_0\}$:

(1) *the set of implementations* $I' \in \{I'\}$ *of the image* I *that represent the given object or scene such that* $I' = \{(x, f(x))\}_{x \in D_f}$ *is the set of points* x *belonging to the definition domain of the implementation of the image* D_f *and the set of values* $f(x)$ *at each point of the definition domain* D_f,
(2) *the semantic and context information on the image* $\{B_0\}$.

The definition domain of the image implementation is a subset of the n-dimensional discrete space Z^n. For plane (two-dimensional) images, $n = 2$.

This lemma can be proved by contradiction given Definition 1.1.

Problems associated with obtaining formal descriptions of images as objects to be analyzed and with forming and choosing recognition procedures can be solved by studying the internal construction, structure, and content of an image as the result of the operations that can be used to construct the image out of subimages and other objects of simpler nature, i.e. primitive elements and objects singled out on the image at various stages of working with it [46].

We assume that the entire process of image processing and recognition, including construction of final formal description, viz. an image model, can be considered as some system of transformations implemented on the image and specified on the equivalence classes [31] that represent assemblies of admissible images.

In DA, we consider three classes of admissible image transformations, viz. procedural transformations $\{\widetilde{O_T}\}$, parametric transformations $\{\widetilde{O_p}\}$, and generating transformations $\{\widetilde{O_G}\}$ [46]. Classes of admissible transformations are connected with the set of structuring elements $\{\tilde{S}\}$ and the set of generating rules $\{\tilde{R}\}$.

Definition 1.3 ([51]). The **parametric transformation** $o_P \in \{\widetilde{O_P}\}$ over the image $\Im$ is such an operation that, when applied to the image implementations $\{\tilde{I}'\} \subseteq \{I'\}$, transforms them into the numerical characteristic p that can be matched with the properties of geometric objects, brightness characteristics or configurations formed

https://avxhm.se/blogs/hill0

at the expense of regular repetitions of geometric objects, and brightness characteristics of the original image. p can be a logical, integer, normal, real, complex-valued value or a vector of fixed dimension with the elements of various nature.

Definition 1.4 ([51]). The **procedural transformation** $o_T \in \{\widetilde{O_T}\}$ **of arity** r over the image $\Im$ is the operation that, when applied to the set of image implementations $\{\tilde{I}'\} \subseteq \{I'\}$, transforms them into some other set of images, some image or its fragment(s).

Definition 1.5 ([51]). The **generating transformation** $o_G \in \{\widetilde{O_G}\}$ is an operation that, when applied to image implementations $\{\tilde{I}'\} \subseteq \{I'\}$, transforms them into some partial representation that (a) depicts specific peculiarities of the image being analyzed and (b) is not (b1) some other image, (b2) a fragment of another image, (b3) a symbolic representation, and (b4) a numerical characteristic.

Definition 1.6 ([46]). The **structuring element** $S \in \{\tilde{S}\}$ is a two-dimensional spatial object, the convolution of which with the image decomposes the image into the system of neighborhoods for it to be analyzed locally. The structuring element is given by parameters that specify its form as well as its numerical and geometrical characteristics.

Definition 1.7 ([51]). The **generating rules** $R \in \{\tilde{R}\}$ are the rules that specify the exact sequence of transformations to be applied to the image I to construct its formal description.

Axiom 1.1 ([46]). *The set of transformations* $\{\tilde{O}\}$ *introduced over information given as the image I is specified by the set of structuring elements* $\{\tilde{S}\}$*, the set of generating rules* $\{\tilde{R}\}$*, and three subsets of transformations, viz.* (1) *procedural transformations* $\{\widetilde{O_T}\}$*,* (2) *parametric transformations* $\{\widetilde{O_P}\}$*, and* (3) *generating transformations* $\{\widetilde{O_G}\}$*.*

We consider the general form of an RCRF procedure that ensures the construction of the mathematical model of the image.

Definition 1.8 ([46]). The **process of construction of the image model** implies that some system of transformations that

ensure the consecutive change of "states" of the image being transformed and correspond to the degree of its current "formalization" is applied to the original image.

Definition 1.9 ([46]). The **formal image model** $M(\mathcal{J})$ is the formal (symbolic) image description constructed as a result of applying **admissible transformations** to $\{\tilde{I}'\} \subseteq \{I'\}$ or to the convolution of $\{\tilde{I}'\} \subseteq \{I'\}$ with structuring element(s) and allowing recognition algorithms not designed for processing spatial information to be applied to it.

Definition 1.10 ([45]). Any element I' of the set of implementations of the original image $\{I'\}$ is called the **I-model** of an image.

As examples of I-models, we can name its gray-scale representation (intensity and grayscale), its binary representation (black and white), its color representation, etc.

In what follows, when we use the image model mathematical object, we mean a formal image model if other interpretation of this notion is not stated explicitly.

For a detailed description of the process of construction of the image model and the definitions of the parametric, generating, and procedural image models, see [45, 46, 51].

When spatial pattern recognition algorithms are used, RCRF procedures are reduced to constructing CSLN representations.

Definition 1.11 ([53]). CSLN representations $SR(\mathcal{J})$, viz. a representation of an image via the so-called "combinatorial structure of local neighborhoods", are spatial information on the image that is a combination of image fragments that (a) reproduce its particular local properties, (b) form local neighborhoods of some order on it, and (c) allow applying "spatial" recognizing algorithms to it.

Definition 1.12 ([53]). A **descriptive algorithmic scheme (DAS)** is a system of transformation that is applied to the original image, implements mathematical methods of image processing, analysis, and recognition, and ensures

(a) constructing the image model or representation that allows applying classical or "spatial" recognition algorithms by successive transformation of "states" of the image being processed that correspond to the degree of its current "formalization",

https://avxhm.se/blogs/hill0

(b) solving the stated image recognition problem by applying recognizing algorithms to the image model/representation constructed according to point (a) of this definition.

Note that in [46] we have introduced DASIR that can be described, according to DAp concepts, using DIA [43, 49, 50, 52]. DASIR implements methods of sequential and/or parallel application of transformations from the set of admissible transformations to the original information from the space of initial data.

In the course of analysis, an image implementation is transformed, by DASIR, from its original form (representation) first into intermediate image representations and then into final image representation that allows a recognizing algorithm to be applied to it. All DASIR applied to original image implementations and then to intermediate image states up to the construction of the image model are combined in the overall DASIR. If we apply the implementation of the overall DASIR to the implementations of the original image, we can construct the image model and/or construct a CSLN representation.

DASIR is a restriction of the DAS notion to the case when CSLN is performed.

1.2.3 *Procedural, parametric, and generating image models*

According to three classes of admissible transformations, by Definition 1.9, we construct three DIM classes, viz. procedural, parametric, and generating ones, respectively.

Suppose the set $\{M_T\}$ is the set of all admissible procedural image models I.

Proposition 1.1 ([51]). *Any* ***T-model*** *of the image* $M_T \in \{M_T\}$ ***generates*** *some* ***implementation*** *of the image* I', *i.e. the I-model of the image* $M_I \equiv I' \in \{M_T\} \equiv \{I'\}$.

According to Definitions 1.4 and 1.9, the procedural model describes an image as a sequence of transformations from the original image or several original images. In the following, we give examples of procedural DIM.

We construct the procedural models M_{T1}, M_{T2}, M_{T3}, and M_{T4} for the image I_1 given in Figure 1.2 and obtained by the OCT-A method

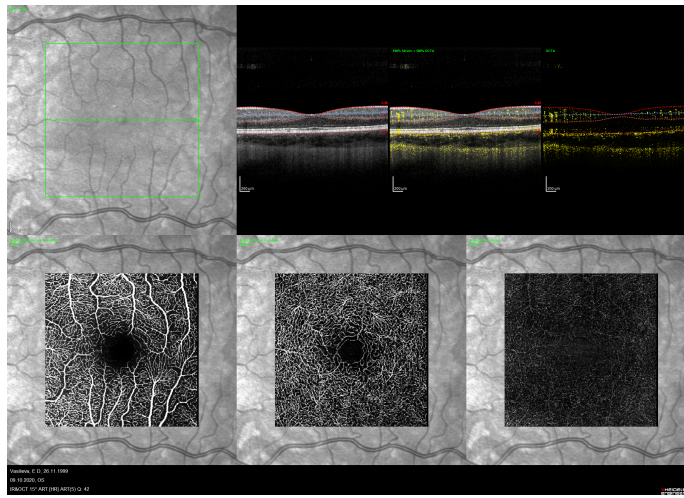


Figure 1.2. Image of the OCT angiogram.

for layer-by-layer analysis of vascular plexus of the retina [30]. The image is the courtesy of the Research Institute of Eye Diseases and is a color picture of the retina tissues decomposed into several layers.

In [30], DAS of recognition of abnormal changes in morphometric characteristics of the fundus on OCT-A images is proposed. To construct the procedural models, we need to perform the following steps, viz. (1) transform the color image into a gray-scale one $t_0 \in \{\widetilde{O_T}\}$, (2) single out the interest zones $t_1 \in \{\widetilde{O_T}\}$, (3) filter the noise on the image $t_2 \in \{\widetilde{O_T}\}$, (4) perform binarization of the image $t_3 \in \{\widetilde{O_T}\}$, (5) remove noise objects by area $t_4 \in \{\widetilde{O_T}\}$, (6) obtain the image of the skeleton of vessels $t_5 \in \{\widetilde{O_T}\}$, and (7) obtain the image of the edges of vessels (perimeter) $t_6 \in \{\widetilde{O_T}\}$.

Then, we can write the procedural models of the image I_1 in the following form:

$$\begin{aligned} M_{T1}(I_1) &= t_0(t_1(I_1)), \\ M_{T2}(I_1) &= t_4(t_3(t_2(M_{T1}(I_1)))), \\ M_{T3}(I_1) &= t_5(M_{T2}(I_1)), \\ M_{T4}(I_1) &= t_6(M_{T2}(I_1)). \end{aligned} \tag{1.1}$$

By Proposition 1.1, any procedural model generates new implementations of the original image given in Figure 1.3.

https://avxhm.se/blogs/hill0

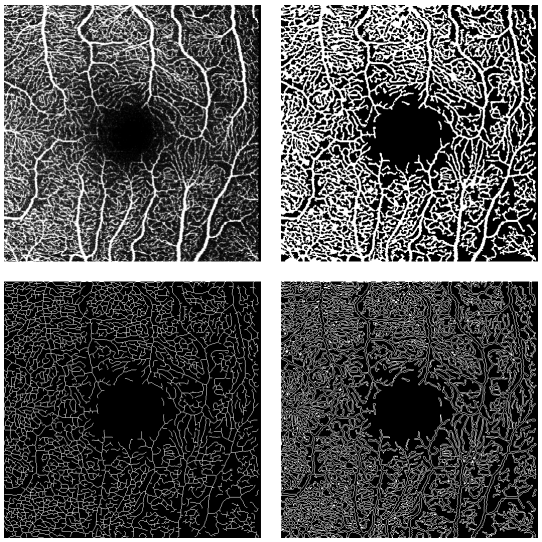


Figure 1.3. Implementations of images of OCT angiograms. In the upper row, left to right, there are implementations generated by $M_{T1}(I_1)$, $M_{T2}(I_1)$. In the lower row, left to right, there are implementations generated by $M_{T3}(I_1)$, $M_{T4}(I_1)$.

We construct another procedural image model $M_{T5}(I_1)$ that corresponds to the localization of the foveolar avascular area (FAA) and ischemic zones on the OCT-A image.

To localize FAA and ischemic zones, we need to apply the following operations to the binary image $M_{T2}(I_1)$, viz. (1) the dilatation operation (image expansion) $t_7 \in \{\widetilde{O_T}\}$, (2) the operation of negative $t_8 \in \{\widetilde{O_T}\}$, and (3) the filtering operation over the FAA area and possible ischemic zones $t_9 \in \{\widetilde{O_T}\}$:

$$M_{T5}(I_1) = t_9(t_8(t_7(M_{T2}(I_1)))). \tag{1.2}$$

Constructed DIMs (1.1) and (1.2) are allowed by any class of spatial recognizing algorithms (for instance, DAEC [26, 36, 37]).

Suppose the set $\{M_P\}$ is the set of all admissible parametric models of the image I.

Proposition 1.2 ([51]). *Any* ***P-model*** *of the image* $M_P \in \{M_P\}$ ***generates*** *a new* ***context and semantic information*** $B_P \subset \{B\}$ *that can be used when constructing image models at a new morphological level.*

According to Definitions 1.5 and 1.10, the parametric model describes an image via a set of numerical characteristics that can be matched with properties of geometric objects, brightness characteristics, or configurations formed at the expense of regular repetitions of geometric objects and brightness characteristics of the original image.

We give an example of parametric DIM.

To analyze OCT-A images, together with experts we formed a list of features that ensure the stated problems of analyzing fundus images to be solved successfully [30]. The features were decomposed into several subgroups, viz. (1) those based on arranging pixels of various brightness, (2) those based on quantitative characteristics calculated using the image on the whole and its particular objects and fragments [8], and (3) a feature based on FAA and ischemic zone localization.

The features falling in some groups are calculated by certain image representation(s) obtained at various stages of image processing, viz. (1) brightness features are calculated by image implementations generated by $M_{T1}(I_1)$, $M_{T2}(I_1)$, (2) the quantitative characteristics are calculated by image implementations generated by $M_{T2}(I_1)$, $M_{T3}(I_1)$, and $M_{T4}(I_1)$, and (3) FAA characteristics are calculated by the image representation generated by $M_{T5}(I_1)$.

Figure 1.4 shows the part of DAS that enables the procedure of forming the features for OCT-A analysis. The scheme represents the sequence of execution of all actions on image processing as well as the relations that show which data are used to form the features. The implementations generated by $M_{T1}(I_1)$, $M_{T2}(I_1)$, $M_{T3}(I_1)$, and $M_{T4}(I_1)$, respectively, are designated on the scheme as OCT-A image, binary image, skeleton image, and perimeter image. The transformations $t_2, t_3, t_4 \in \{\widetilde{O_T}\}$ are designated, respectively, as Gaussian filters to reduce noise, Sauvola binarization, and filter noise by area.

The parametric model of the image $M_{P1}(I_1)$ in Figure 1.4 is designated as OCT-A pixels' brightness. Pixels are preliminary classified based on their brightness.

Together with experts, we chose the optimal thresholds that can be used to decompose pixels into three groups. All black pixels on the binary image (Figure 1.3, upper row, right image) were attributed to the "Not vessels" group. All white pixels on the binary image were compared to the pixels of the original image (Figure 1.3, upper row, left image) — if the original brightness was less than the first

https://avxhm.se/blogs/hill0

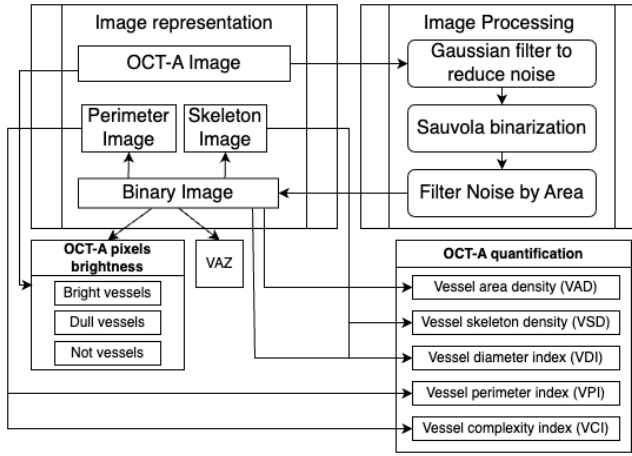


Figure 1.4. Algorithmic scheme of forming the features for analysis of OCT angiograms.

threshold, the pixel was attributed to "Not vessels"; if the brightness was higher than the first threshold yet lower than the second one, this pixel was attributed to the group "Dull vessels"; if the brightness of the pixel was higher than the second threshold, this pixel was treated as a "bright vessel". All pixels were decomposed into three groups. We took the values of the percentage ratio of pixels in the groups with respect to the entire image as the features that describe the image.

We denote the functions of calculating the respective features $f_{nv}, f_{dv}, f_{bv} \in F(\{I\} \to R)$ that translates the set of image implementations into the set of real numbers. The function f_{nv} uses the binary implementation of the original image to calculate the ratio of black pixels with respect to the total number of pixels. The functions f_{dv} and f_{bv} use color and binary implementations of the original image to calculate the relation of dull/bright pixels with respect to the total number of pixels, respectively. We can represent the parametric model $M_{P1}(I_1)$ by the following formula:

$$M_{P1}(I_1) = (f_{nv}(M_{T2}(I_1), \quad f_{dv}(M_{T1}(I_1), M_{T2}(I_1)), \\ f_{bv}(M_{T1}(I_1), M_{T2}(I_1))). \tag{1.3}$$

We consider the construction of the second parametric model $M_{P2}(I_1)$ of the original image given in Figure 1.2. This parametric model of the image in Figure 1.4 is designated as OCT-A quantification. The quantitative evaluation of vascular tortuosity is performed.

We propose to calculate the following parameters:

(1) the vessel area density (VAD) is evaluated by the binary implementation of the original image (Figure 1.3, upper row, right image) as the ratio of the white pixels to the total number of pixels of the image,
(2) the vessel skeleton density (VSD) is evaluated by the implementation of the original image that is responsible for the image of the "skeleton" of vessels (Figure 1.3, lower row, left image) as the ratio of the white pixels to the total number of pixels of the image,
(3) the mean vessel diameter (VDI) is evaluated by the binary implementation and by the implementation responsible for the image of the "skeleton" of vessels of the original image as VAD to VSD ratio,
(4) the density of vascular plexus (VPI) is evaluated by the implementation of the original image that is responsible for the image of the "perimeter" of vessels (Figure 1.3, lower row, right image) as the ratio of the white pixels to the total number of pixels of the image,
(5) the complexity of vascular plexus (VCI) is evaluated by the implementation of the original image responsible for the image of the "perimeter" of vessels (Figure 1.3, lower row, right image) as the ratio of the square of white pixels to the square of the total number of pixels of the image multiplied by 4π.

Evaluation of the vessel area density VAD ensures evaluation of the real density of vascular plexus since it takes into account both the length and diameter of vessels. The bottleneck of such a definition of the parameter is the situation when the vascular bed is decreased and vessels are dilated at the same time, which can lead to false negative results in terms of vascular malformations. VSD provides for the evaluation of the density of vascular plexus and takes into account only the fact of very existence of the vessel since it is only the length of the vessel that is estimated, regardless of its diameter. Since each vessel is represented as one pixel line, big and capillary vessels equally contribute to the quantitative VSD evaluation. Therefore, as compared to VAD, VSD is more sensitive to the decrease of the vascular bed. VDI allows calculating the mean diameter of vessels on the image, yet it fails to depict the change in the vessel density. This parameter is sensitive to changes in vascular distention, yet it

https://avxhm.se/blogs/hill0

does not take into account its length at all. VPI ensures evaluation of the density of vascular plexus and, unlike VSD, it takes into account the vessel perimeter so the estimate includes both the length and diameter of vessels, similar to VAD. VCI ensures evaluation of the complexity of vascular plexus, and the decrease in the area of the vascular bed results in decreased complexity.

All estimates are dimensionless values except for the estimate of complexity of vascular plexus, which is measured in pixels. The parametric model $M_{P2}(I_1)$ is given as follows:

$$\begin{aligned} M_{P2}(I_1) = (1 - f_{nv}(M_{T2}(I_1), \quad & 1 - f_{nv}(M_{T3}(I_1), \\ & (1 - f_{nv}(M_{T2}(I_1))/(1 - f_{nv}(M_{T3}(I_1)), \\ & 1 - f_{nv}(M_{T4}(I_1), \quad n^2(1 - f_{nv}(M_{T4}(I_1))^2/4\pi). \end{aligned} \tag{1.4}$$

We consider the construction of the third parametric model $M_{P3}(I_1)$ of the original image given in Figure 1.2. This parametric model is designated by VAZ in Figure 1.3.

The quantitative estimate of the FAA area and ischemic zones is given as follows:

$$M_{P3}(I_1) = f_{nv}(M_{T5}(I_1)). \tag{1.5}$$

The constructed parametric DIMs are allowed by the overwhelming set of classes of recognition algorithms.

According to Definitions 1.6 and 1.10, a **generating model** is an abstract representation of the image by some generating function and a set of rules of construction of the image from this function.

Suppose, for instance,

(1) images are described by equations of the curves they are composed of,
(2) images are described by indicating the location of principal primitive elements and the relationship between them,
(3) representation of images using various encodings via the alphabet of code characters (code book),
(4) representation of images as Boolean functions, in particular, as disjunctive normal forms,
(5) representation of images as functions of the light field.

Within DAp, we introduce the class of logical transformations (class of descriptive logic transformations, DLT class [28]) that belongs to

the set of generating transformations, ensures transiting from the original image, which is the analysis/recognition object, to the formalized description of this image and obtaining the solution to the recognition problem based on the latter. RCRF DLT transforms the original image into the description given by a disjunctive normal form.

The DLT class possesses the following important advantages, viz.

(1) recognition implies using local and global information reproduced by the image,
(2) the recognizing operators used are computationally efficient,
(3) logical image description is stable with respect to scale and structure deformations.

DLT transformations and special rules of application of transformations are used to construct the special class of generating DIM that describes images as disjunctive normal forms.

We describe the proposed method of reducing the image (generating rules) to the generating model M_G in the form of a standardized algorithmic scheme [28]. To obtain the DNF model of the image, the following transformations from the DLT class need to be sequentially applied to the image, taking into account the internal cycles, viz.

(1) choosing the structuring element $S \in \{S\}$ (see Definition 1.6) and applying it to the image being analyzed,
(2) specifying logical variables (of primitive elements of the image),
(3) detecting the given primitive elements in local neighborhoods of the image singled out using the structuring element (performed for all local neighborhoods of the image) $t_1 \in \{O_T\}$,
(4) constructing the local logical description (logical description of level 0) for the image fragment (logical description of local neighborhood) as an elementary conjunction (performed for all local neighborhoods of the image) $g_0 \in \{O_G\}$,
(5) specifying the node Boolean functions (the Boolean function calculated in several local neighborhoods combined at the node) (logical descriptions of level 1) for the set of adjacent local neighborhoods,
(6) constructing the logical description of level 1 for the image fragment that corresponds to the chosen configuration of adjacent local neighborhoods in the form of DNF, viz. the disjunction of

https://avxhm.se/blogs/hill0

logical descriptions of level 0 (performed for all node Boolean functions of level 1) $g_1 \in \{O_G\}$,

(7) minimizing the node Boolean function of level 1 (performed for all node Boolean functions of level 1),

(8) specifying the node Boolean functions (logical descriptions of level 2) for the sets of adjacent image fragments described by the node Boolean functions of level 1,

(9) constructing the logical description of level 2 for the image fragment corresponding to the chosen configuration of adjacent image fragments described by the node Boolean functions of level 1 in the form of DNF, viz. a disjunction of logical descriptions of level 1 (performed for all node Boolean functions of level 2) $g_2 \in \{O_G\}$,

(10) minimizing the node Boolean functions of level 2 (performed for all node Boolean functions of level 2),

(11) performing steps 8–10 that ensure construction of logical descriptions of image fragments of higher levels in the DNF form $g_3 \in \{O_G\}$,

(12) constructing the logical description in the DNF form for the image on the whole $g_4 \in \{O_G\}$,

(13) minimizing the logical description of the image, viz. constructing the logical image model as a reduced or minimal DNF.

The generating model M_{G1} is given as follows:

$$M_{G1}(I) = \min(g_4(g_3(\min(g_2(\min(g_1(t_1(S \cdot I))))))))).$$

The example of applying this model is given in [28]. The constructed generating DIM reflects information on the content and structure of the image and is allowed by recognizing operators of the class of algorithms based on estimate calculation [28]. Conjunctions included in the DNF model of the image can be used to specify the support sets of the algorithm.

1.3 Descriptive Image Algebras

1.3.1 *Algebraic approach to image processing, analysis, and recognition*

In the 1960s, the need to solve difficult recognition problems as well as the development of structural recognition methods and the

specialized languages for implementation of image processing and analysis methods generated an interest in formal description/models of original data and in the formalization of descriptions of their transformation procedures in the realm of pattern recognition and image analysis.

Significant achievements in this "descriptive" direction of research are described in the works of Ambler *et al.* [2], Evans [17], Kaneff [63], Kirsh [65], Narasimhan [90,91], Rosenfeld [100,101], and Shaw [120].

In the 1970s, Zhuravlev proposed the so-called algebraic approach to recognition and classification problems [138,139], where he defined formalization methods for describing heuristic pattern recognition algorithms and proposed the universal structure of recognition algorithms. This approach forms the basis of the modern mathematical pattern recognition theory.

During the same time period, Grenander stated his theory of patterns [20–23], where he used the algebraic and probabilistic apparatus to consider methods of data representation and transformation in recognition problems in terms of regular combinatorial structures. Both approaches considered the recognition problem in their classical statement and did not touch upon the representation of original data as images.

During the same time period, Pavel [94, 95] proposed a way to use categories in pattern recognition, viz. formal descriptions of pattern recognition algorithms via transformations of original data that preserve their membership in the class.

Within the scientific school of Academician Zhuravlev, significant results in the algebraic direction were obtained by Academician Matrosov [81–84], by Corresponding Member, Russian Academy of Sciences Rudakov [107–113] who elaborated the category approach to solving pattern recognition problems, by Doctor of Physics and Mathematics Mazurov [85–87] who proposed and justified the method of committees, and by Rastrigin and Erenshtein [99] who proposed the method of collective decision rules.

The Russian school of recognition has performed a significant number of researches devoted to algebraic methods of analysis and evaluation of information represented as images and signals, in particular, Doctor of Technical Sciences Labunetc [68], Doctor of Physics and Mathematics Pytiev [97], Doctor of Technical Sciences Sinicyn [121], Doctor of Technical Sciences Furman [18, 19], and Doctor of Physics and Mathematics Chernov [7].

https://avxhm.se/blogs/hill0

Schlesinger's work [119] on two-dimensional grammars is worth noting, where, based on image representations by two-dimensional grammars, he proposed a single statement for those image processing and recognition problems that used to be seen significantly different. The computational complexity of the stated problem is studied in its general statement. Representations are reasonable to be chosen so as to take into account whether it is possible to combine the input information and algorithms of various types.

The idea of multialgorithmic classifiers (MAC) was first proposed and implemented by Zhuravlev in his fundamental work devoted to the algebraic approach to solving recognition and classification problems [138, 139].

Based on Zhuravlev's algebra of algorithms, this approach implies combining algorithms at the stage when algebraic operations are applied to the matrix of responses yielded by particular recognizing algorithms, which ensures construction of the combined matrix of results that represents the result of applying some totality of recognizing algorithms.

Zhuravlev developed and justified algebraic methods for constructing a polynomial over recognition algorithm that solves a recognition problem of arbitrary complexity. It is the form of the algebraic polynomial that defines the specific way of combining particular recognizing algorithms chosen for solving the recognition problem and of combining their results. This approach neither requires nor provides for combining the original data.

In the 1990s, the MAC mechanism was re-invented in English publications and started to be used under the name Multiple Classifiers. Being the subject of many publications, these algorithms are widely used in the automatization of analysis of documents and biometrical data as well as in the automatization of symbol recognition [24, 64, 69, 127–129, 133, 137].

Algorithms are not actually combined in this kind of MAC. In some pre-selected subset of algorithms, the quality of operation is evaluated for each algorithm individually. To obtain the final result, one uses algorithms (the single algorithm, more often) that showed the best result on the learning sample.

Thus far, image analysis and evaluation have accumulated vast experience in applying mathematical methods from various domains of mathematics, computer science, and physics, in particular, algebra,

geometry, discrete mathematics, mathematical logic, probability theory, mathematical statistics, mathematical analysis, mathematical theory of pattern recognition, digital signal processing, and optics.

The principal goal of the algebraic approach to image analysis is to construct the theoretical apparatus that ensures images and their transformations are represented in the form of algebraic structures so that we can use methods borrowed from other realms of mathematics to perform image analysis and recognition.

Algebraization of image analysis is first of all connected with image algebras being defined, constructed, studied, and used. At present, image algebra is understood as a mathematical theory that describes image transformation and analysis in continuous and discrete domains [102].

The idea of creating some unified theory embracing various approaches and operations used in image and signal processing has certain history significantly contributed by the works of Neumann [92, 93] continued by Unger [135], Duff [16], Matheron [80], Ritter [102], Serra [115–117], Sternberg [124–126], and others [11–15, 59–62, 66, 71, 75–78, 98, 122, 123, 130–132, 134].

To a large extent, attempts to create a formal apparatus that would ensure unified and compact representation of image processing and analysis procedures were inspired by practical needs in efficient implementation of algorithmic tools of image processing and analysis on computers with specialized architectures, in particular, cell and parallel ones.

Despite a number of fundamental works [102–105, 115–117] on algebraization of image processing, analysis, and evaluation, one can claim that algebraic theory of image analysis is not yet completely formed.

There are grounds to believe that DAp [27, 29] is one of the leading directions of image analysis algebraization. Algebras of algorithms [27, 29] and DIA introduced by Gurevich within DAp and elaborated by him and his fellow scholars and students [43, 49, 50, 52] are the principal practical algebraization tools.

The bibliographic analysis helped single out the following stages of algebraization in image analysis:

(1) Neumann, the theory of automata and finite-state machines [92, 93],

https://avxhm.se/blogs/hill0

(2) Unger [135], studies of iteration transformations of images in the cellular space,
(3) Batcher [1] and Duff *et al.* [16], a cellular logic array for image processing, special cellular computer architectures for pattern processing,
(4) Rosenfeld [100, 101], digital topology,
(5) Minkowski [89] and Hadwiger [58], arithmetic of neighboring pixels and mathematical morphology,
(6) Zhuravlev, algebra of algorithms [138, 139],
(7) Grenander, pattern theory [20–23],
(8) Pavel, theory of categories in pattern recognition [94, 95],
(9) Matheron [80], Serra [115–117], and Sternberg [126], sequential algebraic theory created especially for image processing and analysis — mathematical morphology,
(10) Sternberg [124], he was the first one to use the term "image algebra",
(11) Maragos [75–78], based on the theory of mathematical morphology, he introduced a new theory combining a big class of linear and nonlinear systems,
(12) Davidson [11], algebra of lattices, also known as the minimum-maximum algebra,
(13) Ritter [102–105], IA,
(14) Dougherty [12–15], homogeneous IA,
(15) Gurevich [25, 29], DAp and DIA,
(16) Gurevich and Yashina, DIA1R [40, 43, 49, 50, 52],
(17) development of mathematical morphology and IA [59–62, 66, 71, 98, 122, 123, 130–132, 134].

1.3.2 *Constructing descriptive image algebras*

1.3.2.1 *Definitions of descriptive image algebras*

Having analyzed the existing algebraic apparatus, we could state the following requirements for the language being created for writing the algorithms to be used to solve image processing and recognition problems, viz.

(1) the new algebra should cover images as analysis and recognition objects,

(2) the new algebra should cover image models, viz. any formal image representations that are objects and sometimes the result of analysis and recognition; introducing image models is a formalization step of the initial data of algorithms,
(3) the new algebra should cover the basic models of image transformation procedures; in addition to using them as operations of the new algebra, it is also convenient to use image transformation procedures as its operands to construct combinations of basic models of procedures.

Based on these requirements, we specified the basic unique features of DIA.

DIA is developed as a descriptive tool, viz. as a language for combining and standardizing the processing procedures of image models and transformations over them.

DIA operands are image representations and models [43] (including original images) and transformations of images, their representations, and models. DIA operations are image analysis and processing transformations, standard algebraic operations, algebraic closures, linear combinations, and superpositions of these operations.

Note that working with image models makes the word "descriptive", i.e. operating with descriptions of images, appear in the new types of algebras.

DIA allows using image transformation procedures as DIA operations as well as its operands to construct combinations of basic models of transformation procedures.

Classical algebra was developed in order to generalize operations over numbers; however, it is not always possible to directly apply the algebra to information represented as images in every problem, and simple interpretation of results is not always possible. There exist many natural image transformations easily interpretable from the user's point (for instance, rotation, compression, stretching, and color inversion) that are difficult to be represented via standard algebraic operations. There arises a need to combine the algebraic apparatus and the set of image analysis and processing transformations.

We remind the DIA definition.

Definition 1.13 ([43]). An algebra is called a **descriptive image algebra** if its operands are either image models (including the fact

https://avxhm.se/blogs/hill0

that both the image and the set of values and characteristics associated with the image can be chosen as the model) or operations over images or all these simultaneously.

DIAs are used to write problems, objects, and transformations dealt with when extracting information from images. When describing algorithmic schemes of formal representation, processing, analysis, and recognition of images using DIA, one gives each element of the scheme and any transformation used in it by the structures constructed by applying DIA operations to the set of DIA operands.

When one uses the structures constructed by applying DIA operations to the set of primitive problems, primitive elements of the image, and basic transformations to specify problems, objects, and transformations considered when extracting information from images, it ensures flexibility and standardization when creating and applying algorithmic schemes of information extraction from images. This approach allows varying methods of solving the subproblem using the operations of image analysis as DIA elements and preserving the scheme of the technique of information extraction from images on the whole.

These procedures are formed and implemented as DAS [53] that represent correct (admissible) statements of the DIA language. The latter are constructed out of operations of processing and transformation of images and other mathematical operations included in the ring of the respective DIA.

Some transformations of image processing, analysis, and recognition can be formally used for mathematical description of the algorithm via DIA; however, they do not have any physical sense specific to image processing and analysis. At the same time, practical applicability of DIA relies on whether the operations used to construct DIA are applicable and implementable in practice. The mathematical and functional/physical interpretation of image analysis and processing operations used as the set of DIA operations was studied.

In mathematics, interpretation is understood as specifying the values (sense) of mathematical expressions (symbols, formulas, etc.). Mathematical objects (sets, operations, expressions, etc.) serve as such values. These values are also called interpretations of the respective expressions.

In our case, we basically speak of algebraic interpretation since DIA is the algebraic language used for mathematical description of

procedures of image processing, analysis, recognition, and understanding implemented using operations of transformation of digital images, their representations, and models.

To characterize the interpretability of DIA operations, we introduced [50, 52] the following notions, viz.

(1) the physical sense of the operation,
(2) physical interpretability in the context of image analysis and processing,
(3) visual interpretability in the context of image analysis and processing,
(4) physical interpretability in the weak sense,
(5) physical interpretability in the strong sense.

These notions are described in more detail in the following section.

For DIA to correspond to the requirements to be met by the "algebra" mathematical object, one needs to introduce restrictions on basic DIA operations. Depending on the algebras used, there arise various DIA classes, viz. DIA with one ring (DIA1R), DIA with two rings (DIA2R), and DIA with multiple rings (DIAMR).

The principal studies of DIA were oriented toward studying DIA1R (see Definition 1.14) that is, by definition, a classical algebra with non-classical operands. We remind that a classical algebra is a ring U that is a finite-dimensional vector space over some field [136].

Definition 1.14 ([44]). The ring U that is a finite-dimensional vector space over some field P is DIA1R if its operands are either image models or operations over images.

The final goal of DIA studies is to obtain sets of complete systems of DIA operands and operations to be used to describe image analysis problems. We use the algebra notion in the strict classical sense in the DIA1R definition because it allows singling out the basic DIA operations for various types of operands.

We believe that the algebra in Birkhoff [6] sense is the most universal one. The **algebra (in Lipson and Birkhoff sense)** is the system $A = [\Phi, F]$, in which (1) $\Phi = \{S_i\}$ is a non-empty set of various types of elements, each of which is called the type of algebra A. The sets S_i are indexed by the set I, i.e. $S_i \in \Phi$ for $i \in I$. (2) $F = \{f_\alpha\}$ is the set of finitary operations, where each f_α is the mapping f_α: $S_{i(1,\alpha)}$ x $S_{i(2,\alpha)}$ x$\cdots$x $S_{i(n(\alpha),\alpha)} \to S_{r(\alpha)}$, for some

https://avxhm.se/blogs/hill0

non-negative integer function $n(\alpha)$, the function i_α: $k \to i(k, \alpha)$ from $n(\alpha) = \{1, 2, \ldots, n(\alpha)\}$ into the set I and $r(\alpha) \in I$. The operations f_α are indexed with respect to the set Ω, i.e. $f_\alpha \in F$ for $\alpha \in \Omega$. In Birkhoff's algebra, two types of universal algebras are singled out, viz. a heterogeneous one (that contains several types of algebras) and a homogeneous one.

Maltsev [74] formed the basis of another general theory bordering with mathematical logic, viz. the theory of algebraic systems. An algebraic system is a set with operations and relationships defined on it. An algebraic system coincides by definition with the homogeneous algebra in the Birkhoff and Lipson sense.

When defining DIA with several rings, we assume one uses the notion of graded algebra and, for two rings, the notion of superalgebra, which are heterogeneous algebras by Birkhoff's definition. By definition, Ritter's standard image algebra [102] is also heterogeneous. We remind that a **graded algebra** [67] is the algebra A, the additive group of which is represented as a (weak) direct sum of the groups $A_i, i = 0, 1, 2 \ldots$, and $A_i A_j \subseteq A_{i+j}$ for any i, j. The additive group of the graded algebra (considered as the module over the ring of integer numbers) is a positive graded module. We remind that a **superalgebra** [73] is a Z_2-graded algebra, i.e. the superspace A over K with the even linear mapping $A \otimes A \to A$. A superspace is a vector space over the field K provided with Z_2-gradation $V = V_{\overline{0}} \oplus V_{\overline{1}}$.

The subsequent DIA unique features are specified by properties of algebras. Figure 1.1 gives the classification of the known algebraic constructions. The DIA1R place in Figure 1.1 can be defined as follows: algebra in Birkhoff's sense/homogeneous algebra/universal algebra/algebra with sigma-operator ring/algebra over an arbitrary ring/sigma-associative commutative ring with unity (linear algebra).

Basic DIA should meet the following requirements, viz. (1) ensure efficient representation of images and their models; (2) ensure representation of DIM as elements of informational techniques of image analysis, recognition, and understanding: (a) ensure generation of basic DIM classes (P-, G-, T-, I-models) and (b) ensure transformation of basic DIM classes; (3) be the elements of the basis of the set of all DIA used for description of image processing, analysis, and recognition; (4) correspond to the principal terms and conditions that specify interpretability of operations of descriptive image algebras.

The mathematical and functional (meaningful/semantic) properties of DIA operations represent significant interest from the point of optimizing how image processing and analysis procedures are chosen and implemented and how the specialized DAS libraries are constructed.

The choice and optimization of operations included in DAS are substantially associated with the specific nature of images as tools of information representation, carriers, and sources.

Functional interpretation of image transformation operations should provide for the connection to be established between image analysis problem and DAS specially designed to solve it. Essentially, this type of interpretation is reduced to establishing the correspondence between meaningful decomposition of the solving process into stages and mathematical DAS operations that ensure implementation of these stages.

Functional interpretation is based on specific informational image properties considered above (Section 1.2.1).

Image processing and recognition process, including construction of the image model, can be treated as some system of transformations implemented on the image.

This means that to ensure the efficiency of synthesis of the model and recognition, one should choose the "content" (function) of image transformation operations, depending on what image representation should be obtained using yet another transformation. In its turn, such choice should be based both on the analysis of mathematical characteristics of the operation and on the analysis of its functional purpose, in other words, semantic aspects of the operation, i.e. its content, detection of the physical equivalent, and functional heuristics that underlie it.

Since not all mathematical operations have the direct physical equivalent, constructing efficient image analysis DAS raises the issue of interpretation of the operations that fill DAS. If we study this issue, we arrive at choosing and studying domains of interpretation of DIA operations.

Thus, interpretation is treated as the transition from a meaningful description of the operation to its mathematical or algorithmic implementation. As a result, practical applicability of operations unfolds in the context of a more general notion of interpretability.

https://avxhm.se/blogs/hill0

The problem statement and initial results are given in [52]. In [50], the method and tools of formalization of types of interpretability of image analysis and processing operations are given as well as 18 examples of operands with operations introduced on them, for which interpretability is studied, and the example of constructing DAS in the DIA1R language for solving an applied problem that illustrates another type of interpretation as compared to 18 examples described, viz. the algorithmic type.

In the following, we give the results related to interpretability of DIA1R operations and examples of interpretability domains for some types of operations.

Definition 1.15 ([50, 52]). The **physical sense of the operation** is understood as a meaningful description of the process of transformation of the initial image(s) into the final image(s) or description of the statement according to the original image of some set of characteristics.

To preserve the logic of further consideration, we recall some notions of DAp associated with the description of the image processing and analysis process and complementary definitions of the image model/representation [51].

In the course of image processing and analysis, some system of transformations is applied to the original image so that the "phase states" of the image being transformed that correspond to the degree of its current "formalization" succeed each other. The set of admissible image representations is defined as the set of "phase states" of the image.

The system of transformations is given by DASIR written according to DAp concepts using DIA. DASIR reflects the ways of transformations from the set of transformations being applied consecutively and/or parallelly to the original information from the space of initial data. The set of admissible DASIR is specified as the set of "phase states" of DASIR.

For recognition algorithms to be applicable to the constructed image descriptions, one needs to implement the constructed DASIR (give the particular transformations from the fixed DIA and parameters of transformations included in the schemes) and apply the implemented schemes to original data, i.e. construct image representations and models. IFS is specified that includes both "phase" states of the

image and "phase" states of DASIR. The image representation is any element of the set of "states" of the image in IFS, excluding the "image model" and "image implementation" objects. For a more detailed description of IFS, see [46].

Definition 1.16 ([50, 52]). An operation over image(s) or their fragments or over image model(s) or image representations is called **physically interpretable in the context of image analysis and recognition** if

(1) the result of its application is an image or its fragments,
(2) the result of its application is an image representation or model that can be used to restore semantically essential geometric objects, brightness characteristics, and configurations formed by regular repetitions of geometric objects and brightness characteristics of the original image,
(3) the result of its application is image characteristic(s) that can be unambiguously matched with properties of geometric objects, brightness characteristics, and configurations formed by regular repetitions of geometric objects and brightness characteristics of the original image.

Definition 1.17 ([50, 52]). An operation over some objects is called **visually interpretable in the context of image analysis and recognition** if it results in an image or images that can be used to restore the one-to-one correspondence between semantically essential geometric objects, brightness characteristics, and configurations formed by regular repetitions of geometric objects and brightness characteristics on the obtained image(s) and on the original objects.

Proposition 1.3 ([50, 52]). *A visually interpretable operation is always physically interpretable.*

Corollary ([50,52]). *If an operation is physically non-interpretable, this operation is also visually non-interpretable.*

One can distinguish the physical interpretability in the strong and weak sense.

Definition 1.18 ([50, 52]). An operation is called **physically interpretable in the strong sense** if it is also visually interpretable.

https://avxhm.se/blogs/hill0

Definition 1.19 ([50, 52]). An operation is called **physically interpretable in the weak sense** if it is physically interpretable but is not visually interpretable.

Visually interpretable operations are, for instance, image rotation, image shift, increasing image contrast, increasing image brightness, decreasing noise in the image, image smoothing, contour detection on the image, and other image processing operations. As an example of visually interpretable operations, we can also name operations of constructing an image using some given rule out of the set of original objects, for instance, restoring an image from equations that give the form of the image.

Physically interpretable operations include some operations of constructing image representations and models and such operations over images as calculating an image histogram or calculating values of statistical features of the image.

Proposition 1.4 ([50, 52]). *An operation is* ***physically non-interpretable in the context of image analysis and recognition*** *if*

(1) *the operation is not applicable to image(s), i.e. images, image models, image representations, and image fragments cannot be its operands,*
(2) *applying the operation to image(s) constructs an image model that cannot be used to restore semantically essential geometric objects, brightness characteristics or configurations formed by regular repetitions of geometric objects, and brightness characteristics on the original image,*
(3) *as a result of applying the operation, the image is used to calculate characteristics that cannot be unambiguously matched to properties of geometric objects, brightness characteristics or configurations formed by regular repetitions of geometric objects, and brightness characteristics on the original image.*

1.3.2.2 *Method of constructing descriptive image algebras*

The DIA construction method is based on specifying the operands, operations, restrictions, and relationships needed to construct algebras that define them. To make DIA suitable for application, we also study interpretability of the chosen operations.

We assume that the image recognition problem contains

(1) the problem statement, including the goals of image processing, analysis, and recognition within this problem,
(2) the problem's hypotheses — (a) prohibited and impossible order relations, partial order relations, and other conditions and restrictions that reflect physical and logical organization of the real world; (b) requirements on the results of solving the problem; (c) restrictions on using the resources necessary for solving the problem (computational complexity and time); and (d) recommendations regarding the solution methods,
(3) original data.

The DIA construction method includes the following steps:

Step 1: According to the stage of the image recognition problem, choose a set (field) for the algebra to be given.

Step 2: According to the stage of the image recognition problem, give the operands of the set considered as a potential DIA.

Step 3: According to the stage of the image recognition problem, give the operations of the set considered as a potential DIA.

Step 4: From the transformations given in Step 3, assign the operations needed to construct the algebra that specifies DIA (in particular, for DIA1R, the operations of addition, product, and multiplication by the field elements — these operations define a ring over a field).

Step 5: Study interpretability of the chosen DIA operations.

Step 6: Introduce the necessary restrictions on the chosen field, operands, and operations of the given set (Steps 1–4).

Step 7: State and prove the theorem on the existence of DIA of the given form, viz. the field from Step 1, the operands of the algebra from Step 2, the operations of the algebra from Step 4 that satisfy the restrictions introduced in Step 6 and interpreted according to Step 5.

The necessary and sufficient conditions of DIA1R construction are based on properties of classical algebra.

Theorem 1.1. *Suppose P is a field of any nature*; *U is a set of elements of the following nature* (1) *image implementations*, (2)

https://avxhm.se/blogs/hill0

image transformations, (3) *image representations*, (4) *image models*; Θ*-operations of addition* $(+)$, *product* $(\cdot)$, *and multiplication by an element of the field* (α) *chosen from the set of operations*, *viz.* (1) *standard algebraic operations*, (2) *transformations of image analysis and processing*, *and* (3) *algebraic closures*, *linear combinations and superpositions of these operations. The mathematical construction* $(P,\ U,\ \Theta)$ *is* ***DIA1R*** *if and only if the elements of the set* U *with the set of operations* Θ *and field* P *introduced on it satisfy the following properties of the classical algebra*: I. *The properties of the ring* $U\ (a, b, c \in U)$: (1) $\forall a, b \in U, \exists!\ (a + b) \in U$: $(a)\ a + (b + c) = (a + b) + c$, $(b)\ a + b = b + a$, $(c)\ \exists\ 0 \in U, \forall a \in U, a + 0 = a$, $(d)\ \forall a \in U, \exists(-a), a + (-a) = 0$; (2) $\forall a, b \in U, \exists!\, (a \cdot b) \in U$: $(a)\ a \cdot (b \cdot c) = (a \cdot b) \cdot c$, $(b)\ (\alpha \cdot a + \beta \cdot b) \cdot c = \alpha \cdot a \cdot c + \beta \cdot b \cdot c$ II. *The properties of the vector space* $(\forall \alpha \in P, a \in U$: $\alpha \cdot a \in U)$: (1) $\alpha \cdot (\beta \cdot a) = (\alpha \cdot \beta) \cdot a$, (2) $(\alpha + \beta) \cdot a = \alpha \cdot a + \beta \cdot a$, *and* (3) $\alpha \cdot (a + b) = \alpha \cdot a + \alpha \cdot b$.

The proof of this theorem follows from the DIA1R definition and the definition of classical algebra.

To construct DIA1R, at Step 4 of the method, we choose the following operations from the given ones — the operation of addition of elements of the given set, the operation of the product of elements of the given set, and the operation of multiplication of the element of the set by the element of the chosen field. The proof of the theorem on the existence of DIA1R of the form given at Step 7 relies on the necessary and sufficient conditions of DIA1R construction (see Theorem 1.1).

We give the interpretation of the DIA1R construction method in the following.

1.4 Classes of Descriptive Image Algebras with One Ring

1.4.1 *Descriptive image algebras with one ring over images*

According to Definitions 1.13 and 1.14 and Theorem 1.1, DIA1R are constructed over the following operands, viz. (1) image implementations, (2) image transformations, (3) image representations,

and (4) image models. To study various DIA1R classes, one uses either standard algebraic operations or image analysis and processing transformations or algebraic closures, linear combinations, and superpositions of these operations.

To construct DIA1R over images, we need to apply some transformations to image implementations according to the context and semantic information on the image. Image transformations should satisfy the properties of the algebra [136] and, first of all, should be closed with respect to the original data.

Since image processing [39] is a transformation of one image into another using the operations of compression, restoration, improvement of visual quality, quantization, and filtering, we can say that image processing operations are closed operations with respect to the set of images, unlike operations of image understanding, analysis, classification, and recognition. Therefore, to construct DIA1R over images, one applies transformations (as operations of the algebra) used to solve primitive image processing problems [39]. To choose image processing transformations to construct examples of DIA1R, one uses the image analysis and processing thesaurus [3,4] that includes, among other things, classifications of image processing methods, operations, and problems.

Ritter's standard image algebra (SIA) is the most detailed existing approach to describing operations over images in terms of algebraic operations over basic operations with simple structure. It allows formalizing image processing algorithms at various levels both to make their clear representation and to adapt them to parallel and distributed architectures.

Creating the new specialized DIA1R leads to studying the already existing SIA in order to see if it is possible to use the detailed description of SIA operations to generate DIA1R.

Ritter [102] introduces SIA operations in our sense over image implementations, i.e. any image I is described by its implementation $I' = \{(x, f(x))\}_{x \in D_f}$ — the set of points x that belong to the definition domain D_f of the image implementation and the set of values $f(x)$ at each point of the definition domain D_f. Semantic and context information on the image is taken equivalent to the empty set $B_0 = \emptyset$. For the detailed study of interpretation of SIA operations, see [102].

https://avxhm.se/blogs/hill0

1.4.1.1 *DIA1R1.1 over image implementations and operations of Ritter's standard IA*

We consider DIA1R over image implementations based on operations of addition, product, and multiplication by an element of the field of real numbers chosen from the operations of Ritter's standard IA: **DIA1R1.1**. This DIA1R is interesting due to the method of specifying image implementation on the fixed definition domain of the image with various sets of image values at each point of the image definition. To construct DIA1R over image implementations of such form, we introduce restrictions on the set of image values.

Step 1: Suppose the algebra under construction is given over the field of real numbers R (suppose $\alpha \in R$).

Step 2: Let the operands of the algebra to be constructed be the images specified on the fixed set X with various fixed ranges of values $F_i, i = 1, 2, \ldots$: $I = \{(x, f(x)), x \in X, f(x) \in F_i\}$. Let $I_1 = \{(x, a(x)), x \in X, a(x) \in F_1\}$, $I_2 = \{(x, b(x)), x \in X, b(x) \in F_2\}$ designate any two elements of the set of the given operands.

The example of the function $f(x)$ and sets F_i and X that give the image includes $X = [0..N, 0..N]$, which is a square in the two-dimensional plane, $f(x) = (r(x), g(x), b(x))$, where $r(x), g(x)$, and $b(x)$ are three color components — red, green, and blue, respectively, and the sets $F_i \subset R^3$, which are the sets that bound the color gamma of images, for instance, they correspond to the gamma of a black-and-white image, red image, green image, yellow image, etc.

Step 3: On the set of the chosen operands, we give the following operations of the standard IA:

$$I_1 + I_2 = \{(x, a(x) + b(x)), x \in X\}, \tag{1.6}$$

$$I_1 \cdot I_2 = \{(x, a(x) \cdot b(x)), x \in X\}, \tag{1.7}$$

$$\alpha \cdot I = \{(x, \alpha \cdot f(x)), x \in X\}. \tag{1.8}$$

Step 4: Operation (1.6) is the operation of addition of elements of the given set, operation (1.7) is the operation of product of elements of the given set, and operation (1.8) is the operation of multiplication of the element of the set by an element of the chosen field.

Step 5: We study interpretability of the chosen operations.

The physical sense of operation (1.6): The operation is matched to the point-by-point addition of two images.

The physical sense of operation (1.7): The operation is matched to the point-by-point product of two images.

Proposition 1.5. *Operations of point-by-point addition* (1.6) *and product* (1.7) *of two images are* ***physically interpretable in the weak sense.***

Proof.

1. By Definition 1.16, an operation is physically interpretable if the result of its application is an image.
2. By Definition 1.17, this operation is not visually interpretable since when it is applied to arbitrary images, the form of the visual representation of the result cannot be determined beforehand.
3. By Definition 1.19, a physically interpretable yet visually non-interpretable operation is physically interpretable in the weak sense.

Q.E.D.

The physical sense of operation (1.8): The operation is matched to the multiplication of an image by a number.

Proposition 1.6. *Operations of form* (1.8) *of multiplication of an image by a number are* ***physically interpretable in the strong sense.***

Proof.

1. By Definition 1.16, an operation is physically interpretable if the result of its application is an image.
2. By Definition 1.17, this operation is visually interpretable since when it is applied to an image, its brightness changes visually.
3. By Definition 1.18, a physically and visually interpretable operation is physically interpretable in the strong sense.

Q.E.D.

Step 6: We introduce the necessary restrictions on the chosen field, operands, and operations of the given set (Steps 1–4).

Suppose the set of sets $\{F_i\}_1^\infty$ is given. We bound the values of the sets by the n-dimensional real plane.

https://avxhm.se/blogs/hill0

We introduced the operation of addition of two elements from F_i, $F_j \subset R^n$: i, $j = 1, 2, \ldots, \forall a \in F_i$, $b \in F_j$: $\exists!$ $a + b \in F_k$, $k = 1, 2 \ldots, F_k \subset R^n$ that possesses the following properties ($\forall a \in F_i$, $b \in F_j$, $c \in F_y$, i, j, $y = 1, 2, \ldots$):

$$a + (b + c) = (a + b) + c, \tag{1.9}$$

$$a + b = b + a, \tag{1.10}$$

$$\forall a \in F_i, \exists\, 0 \in F_i : a + 0 = a, \tag{1.11}$$

$$\forall i : \forall a \in F_i, \exists\, (-a) \in F_i : a + (-a) = 0. \tag{1.12}$$

We introduced the operation of the product of two elements from F_i, F_j: i, $j = 1, 2, \ldots, a \in F_i$, $b \in F_j$: $\exists!$ $a \cdot b \in F_k$, $k = 1, 2 \ldots, F_k \subset R^n$ that possesses the following property ($\forall a \in F_i$, $b \in F_j$, $c \in F_y$, i, j, $y = 1, 2, \ldots$):

$$a \cdot (b \cdot c) = (a \cdot b) \cdot c. \tag{1.13}$$

On the set $F_i (i = 1, 2, \ldots)$, we introduced the operation of multiplication by an element of the field of real numbers R: $\forall \alpha \in R$, $a \in F_i$: $\exists!$ $\alpha \cdot a \in F_i (i = 1, 2, \ldots)$ that possesses the following properties ($\forall a \in F_i$, $b \in F_j$, $c \in F_y$, i, j, $y = 1, 2, \ldots, \forall \alpha, \beta \in R$):

$$(\alpha \cdot a + \beta \cdot b) \cdot c = \alpha \cdot a \cdot c + \beta \cdot b \cdot c, \tag{1.14}$$

$$\alpha \cdot (\beta \cdot a) = (\alpha \cdot \beta) \cdot a, \tag{1.15}$$

$$(\alpha + \beta) \cdot a = \alpha \cdot a + \beta \cdot a, \tag{1.16}$$

$$\alpha \cdot (a + b) = \alpha \cdot a + \alpha \cdot b \tag{1.17}$$

If all $F_i \equiv F \subset R^n (i = 1, 2, \ldots)$, the set F with the operations introduced in such a way is an algebra over the field of real numbers.

Step 7: We state and prove the theorem on the existence of DIA1R of the given form.

Theorem 1.2 ([40]). *Suppose R is the field of real numbers, $I = \{(x, f(x)),\ x \in X, f(x) \in F\}$ ($F \in \{F_i\}_1^\infty$ (the set of sets $\{F_i\}_1^\infty$ satisfies properties (1.9)–(1.17)), F is the set of values taken by the image I on the set X) are the elements of the set $U_1; I_1 = \{(x, a(x)), x \in X, a(x) \in F_1\}$, $I_2 = \{(x, b(x)), x \in X, b(x) \in F_2\}$, and $(F_1, F_2 \in \{F_i\}_1^\infty)$ are any two elements of the set U_1. We introduce the operation of addition of two images I_1 and I_2 of form (1.6),*

the operation of the product of two images I_1 *and* I_2 *of form* (1.7), *and the operation of multiplication of the image* I *by the element of the field of real numbers* $\alpha \in R$ *of form* (1.8). *Then, the* ***set*** $\mathbf{U_1}$ *with the operations of addition, product, and multiplication by an element of the field of real numbers introduced on it is* ***DIA1R over the field of real numbers.***

Proof. The proof is based on verifying the properties of the classical algebra on the set U_1 with the operations of addition, product, and multiplication by an element of the field of real numbers introduced on it.

We check the properties of the algebra. R is the field of real numbers. We check the properties of the ring U_1 $(I_1, I_2, I_3 \in U_1)$.

Suppose $\mathbf{I_1 = \{(x, a(x)), x \in X, a(x) \in F_1\}, I_2 = \{(x, b(x)), x \in X, b(x) \in F_2\}, I_3 = \{(x, c(x)), x \in X, c(x) \in F_3\}.}$

1. We check the properties of the addition operation.
   $\forall I_1, I_2 \in U_1$, $\exists!\ (I_1 + I_2) \in U_1$.
   (a) $I_1 + (I_2 + I_3) = I_1 + \{(x, b(x) + c(x)),\ x \in X\} = \{(x, a(x) + (b(x) + c(x))),\ x \in X\}$; $(I_1 + I_2) + I_3 = \{(x, a(x) + b(x)), x \in X\} + I_3 = \{(x, (a(x) + b(x)) + c(x)),\ x \in X\} =$ {by property 1.9} $= \{(x, a(x)+(b(x)+c(x))), x \in X\}$; i.e. the property of associativity of the addition operation is **proved** to hold: $I_1 + (I_2 + I_3) = (I_1 + I_2) + I_3$.
   (b) $I_1 + I_2 = \{(x, a(x) + b(x)), x \in X\}$; $I_2 + I_1 = \{(x, b(x) + a(x)), x \in X\} =$ {by property 1.10} $= \{(x, a(x) + b(x)), x \in X\}$; i.e. the property of commutativity of the addition operation is **proved** to hold: $I_1 + I_2 = I_2 + I_1$.
   (c) $O = \{(x, 0), x \in X\}$; $I_1 + O = \{(x, a(x) + 0), x \in X\} =$ {by property 1.11} $= \{(x, a(x)), x \in X\} = I_1$; the property of existence of 0 is **proved** to hold: $\exists\ O \in U, \forall I_1 \in U_1$, $I_1 + O = I_1$.
   (d) $(-I_1) = \{(x, -a(x)), x \in X\}$; $I_1 + (-I_1) = \{(x, a(x) - a(x)), x \in X\} =$ {by property 1.12} $= \{(x, 0), x \in X\} = O$; the property of existence of the inverse element is **proved** to hold: $\forall I_1 \in U, \forall (-I_1) \in U_1, I_1 + (-I_1) = O$.
2. We check the properties of the product operation.
   $\forall\ I_1,\ I_2 \in U_1$, $\exists!\ (I_1 \cdot I_2) \in U_1$.
   (a) $I_1 \cdot (I_2 \cdot I_3) = I_1 \cdot (\{(x, b(x) \cdot c(x)), x \in X\}) = \{(x, a(x) \cdot (b(x) \cdot c(x))), x \in X\}$; $(I_1 \cdot I_2) \cdot I_3 = (\{(x, a(x) \cdot b(x)), x \in X\}) \cdot I_3 =$

https://avxhm.se/blogs/hill0

$\{(x,(a(x)\cdot b(x))\cdot c(x)), x \in X\}$ = {by property 1.13} = $\{(x, a(x)\cdot(b(x)\cdot c(x))), x \in X\}$; i.e. the property of associativity of the product operation is **proved** to hold: $I_1 \cdot (I_2 \cdot I_3) = (I_1 \cdot I_2) \cdot I_3$.

3. We check the properties of multiplication by an element of the field.
   Suppose $\alpha, \beta \in R$; $(\alpha \cdot I_1 + \beta \cdot I_2) \cdot I_3 = (\{(x, \alpha \cdot a(x) + \beta \cdot b(x)), x \in X\}) \cdot I_3 = \{(x, (\alpha \cdot a(x) + \beta \cdot b(x)) \cdot c(x)), x \in X\} =$ {by property 1.12} $= \{(x, \alpha \cdot a(x) \cdot c(x) + \beta \cdot b(x) \cdot c(x)), x \in X\}$; $\alpha \cdot I_1 \cdot I_3 + \beta I_2 \cdot I_3 = \{(x, \alpha \cdot a(x) \cdot c(x)), x \in X\} + \{(x, \beta \cdot b(x) \cdot c(x)), x \in X\} = \{(x, \alpha \cdot a(x) \cdot c(x) + \beta \cdot b(x) \cdot c(x)), x \in X\}$; i.e. the following property is **proved** to hold: $(\alpha \cdot I_1 + \beta \cdot I_2) \cdot I_3 = \alpha \cdot I_1 \cdot I_3 + \beta \cdot I_2 \cdot I_3$.
4. We check that the properties of the vector field hold ($\forall \alpha \in P$, $I \in U_1$: $\alpha \cdot I \in U_1$).
   (a) $\alpha \cdot (\beta \cdot I) = \alpha \cdot \{(x, \beta \cdot f(x)), x \in X\} = \{(x, \alpha \cdot (\beta \cdot f(x))), x \in X\} =$ {by property 1.15} $= \{(x, \alpha \cdot \beta \cdot f(x)), x \in X\}$; $(\alpha \cdot \beta) \cdot I = \{(x, \alpha \cdot \beta \cdot f(x)), x \in X\}$; i.e. the property $\alpha \cdot (\beta \cdot I) = (\alpha \cdot \beta) \cdot I$ is **proved**.
   (b) $(\alpha+\beta)\cdot I = \{(x, (\alpha+\beta)\cdot f(x)), x \in X\} =$ {by property 1.16} $= \{(x, \alpha \cdot f(x) + \beta \cdot f(x)), x \in X\}$; $\alpha \cdot I + \beta \cdot I = \{(x, \alpha \cdot f(x)), x \in X\} + \{(x, \beta \cdot f(x)), x \in X\} = \{(x, \alpha \cdot f(x) + \beta \cdot f(x)), x \in X\}$; i.e. the property $(\alpha + \beta) \cdot I = \alpha \cdot I + \beta \cdot I$ is **proved**.
   (c) $\alpha \cdot (I_1 + I_2) = \alpha \cdot \{(x, a(x) + b(x)), x \in X\} = \{(x, \alpha \cdot (a(x) + b(x))), x \in X\} =$ {by property 1.17} $= \{(x, \alpha \cdot a(x) + \alpha \cdot b(x)), x \in X\}$; $\alpha \cdot I_1 + \alpha \cdot I_2 = \{(x, \alpha \cdot a(x)), x \in X\} + \{(x, \alpha \cdot b(x)), x \in X\} = \{(x, \alpha \cdot a(x) + \alpha \cdot b(x)), x \in X\}$; i.e. the property $\alpha \cdot (I_1 + I_2) = \alpha \cdot I_1 + \alpha \cdot I_2$ is **proved**.

All properties of the ring, field, and vector space hold. The set U_1 with the operations of addition, product, and multiplication by an element of the field of real numbers introduced on it is an algebra over the field of real numbers. By the nature of the elements of the set U_1, this algebra *is DIA1R* (*see Theorem* 1.1) Q.E.D.

1.4.1.2 *DIA1R1.2 over image implementations and special operations of Ritter's standard IA*

In [43], we considered one more specialized version of DIA1R over image implementations using operations of Ritter's standard IA, viz.

DIA1R1.2. In this algebra, we demonstrated the operation of superposition of two functions used as the operation of product of two DIA1R elements. As the operands of the algebra to be constructed, we chose images given on the fixed set X with the range of values X: $I = \{(x, f(x)), x \in X, f(x) \in X\}$. As an example of such functions and sets that give an image, we can name the set $X = R^3$; the function $f(x)$ of the image I of the three-dimensional scene is specified at the point of the space (x, y) and takes into account the distance from the source of light l, i.e. $x = (x, y, l)$; the function $f(x)$ is $(r(x), g(x), b(x))$, where $r(x), g(x), b(x)$ are the three color components, viz. red, green, and blue, respectively.

Suppose $I_1, I_2 \in (X)^X$.

We introduce the superposition operation

$$I_1(I_2) = \{(x, a(b(x)), x \in X\}. \tag{1.18}$$

The physical sense of the operation: Specifying one image on the set given by another image.

Proposition 1.7. *The operation of superposition of two images of type* (1.18) *is* ***physically interpretable in the weak sense.***

Existence of DIA1R of the given type is confirmed by the following theorem.

Theorem 1.3 ([43]). *Suppose R is a field of real numbers; the images $I = \{(x, f(x)), x \in X, f(x) \in X\}$ constructed on the same original set X with the set of values X coinciding with the original one are the elements of the set U_2; $I_1 = \{(x, a(x)), x \in X, a(x) \in X\}$, $I_2 = \{(x, b(x)), x \in X, b(x) \in X$ are any two elements of the set U_2. One introduces the operation of addition of two images I_1 and I_2 of form* (1.6), *the operation of product of two images I_1 and I_2 of form* (1.18), *and the operation of multiplication of the image I by an element of the field of real numbers $\alpha \in R$ of form* (1.8). *Then, the* ***set U_2*** *with the operations of addition, product, and multiplication by an element of the field of real numbers introduced on it is* ***DIA1R over the field of real numbers.***

The proof is based on verifying the properties of the classical algebra of the set U_2 with the operations of addition, product, and multiplication by an element of the field of real numbers introduced on it (Theorem 1.1).

https://avxhm.se/blogs/hill0

1.4.1.3 *DIA1R1.3 over color images*

We demonstrate the specialized versions of DIA1R with operations of image processing that differ from those described by Ritter [103].

Note that the principal number of operations of image processing is applied to one image rather than two, thus making the choice of binary operations of addition and product used to construct DIA1R rather limited. Steps of the method in this and subsequent examples in this chapter are omitted, with only statements of the fundamental theorems of existence or non-existence of DIA1R of the given type presented and interpretability of the chosen operations studied.

We consider **DIA1R1.3** over color images. Existence of this algebra is confirmed by the following theorem.

Theorem 1.4 ([49]). *Suppose R is a field of real numbers; the elements of the set U are $I = \{((x,y),(r(x,y),g(x,y),b(x,y))\}_{(x,y)\in X}$, where the functions $r(x,y),g(x,y),b(x,y)$ take integer values from the set $[0\cdots M-1]; M = 256$ is the value of the maximal brightness of the color components; X is the set of pixels the images are given at; $I_1 = \{((x,y),(r_1(x,y),g_1(x,y),b_1(x,y)))\}_{(x,y)\in X}$, $I_2 = \{((x,y),(r_2(x,y),g_2(x,y),b_2(x,y)))\}_{(x,y)\in X}$, where $r_1(x,y),g_1(x,y),b_1(x,y),r_2(x,y),g_2(x,y),b_2(x,y) \in [0\cdots M-1]$ are any elements of the set U and they take integer values. We introduce*

- *the operation of addition of two images I_1 and I_2*

$$\begin{aligned} I_1 + I_2 = \{((x,y),&((r_1(x,y)+r_2(x,y)) \bmod M,\\ &(g_1(x,y)+g_2(x,y)) \bmod M,\\ &(b_1(x,y)+b_2(x,y)) \bmod M)\}_{(x,y)\in X}, \end{aligned} \tag{1.19}$$

- *the operation of product of two images I_1 and I_2*

$$\begin{aligned} I_1 \cdot I_2 = \{((x,y),&((r_1(x,y)\cdot r_2(x,y)) \bmod M,\\ &(g_1(x,y)\cdot g_2(x,y)) \bmod M,\\ &(b_1(x,y)\cdot b_2(x,y)) \bmod M)\}_{(x,y)\in X}, \end{aligned} \tag{1.20}$$

- *the operation of multiplication of the image I by an element of the field of real numbers $\alpha \in R$*

$$\begin{aligned} \alpha \cdot I = \{((x,y),&([\alpha r(x,y) \bmod M],[\alpha g(x,y) \bmod M],\\ &[\alpha b(x,y) \bmod M]))\}_{(x,y)\in X}. \end{aligned} \tag{1.21}$$

Then, the *set* U with the operations of addition, product, and multiplication by an element of the field of real numbers introduced on it is *DIA1R over the field of real numbers*.

The proof is based on verifying the properties of the classical algebra of the set U with the operations of addition, product, and multiplication by an element of the field of real numbers introduced on it (Theorem 1.1).

The physical sense of operation (1.19): The operation is matched to point-by-point addition modulo M of color components of two color images.

The physical sense of operation (1.20): The operation is matched to point-by-point product modulo M of color components of two color images.

The physical sense of operation (1.21): The operation of multiplication modulo M of an element of the field by the color image.

Proposition 1.8. *Introduced operations* (1.19) *and* (1.20) *of point-by-point addition and product of two color images are* ***physically interpretable in the weak sense.***

Proposition 1.9. *Introduced operation* (1.21) *of point-by-point multiplication of the color image by an element of the field of real numbers is* ***physically interpretable in the strong sense.***

1.4.1.4 *DIA1R1.4 over gray-scale images*

We consider **DIA1R1.4** over gray-scale images. This algebra is introduced similar to the previous one. Existence of this algebra is confirmed by the following theorem.

Theorem 1.5. *Suppose* R *is the field of real numbers*; *the elements of the set* V *are* $J = \{((x,y), gray(x,y))\}_{(x,y)\in X}$, *where the function* $gray(x,y)$ *takes integer values from the set* $[0 \cdots M-1]$; $M = 256$ *is the value of the maximal brightness of the color components*; X *is the set of pixels the images are given at*; $J_1 = \{((x,y), gray_1(x,y))\}_{(x,y)\in X}, J_2 = \{((x,y), gray_2(x,y))\}_{(x,y)\in X}$ *are the elements of the set* V, *where* $gray_1(x,y), gray_2(x,y) \in [0 \cdots M-1]$ *and take integer values. We introduce*

https://avxhm.se/blogs/hill0

- *the operation of addition of two images* J_1 *and* J_2

$$J_1 + J_2 = \{((x,y), (\mathrm{gray}_1(x,y) + \mathrm{gray}_2(x,y)) \bmod M)\}_{(x,y)\in X}, \tag{1.22}$$

- *the operation of product of two images* J_1 *and* J_2

$$J_1 \cdot J_2 = \{((x,y), (\mathrm{gray}_1(x,y) \cdot \mathrm{gray}_2(x,y)) \bmod M)\}_{(x,y)\in X}, \tag{1.23}$$

- *the operation of multiplication of the image* J *by the element of the field of real numbers* $\alpha \in R$

$$\alpha \cdot J = \{((x,y), [\alpha \cdot \mathrm{gray}(x,y) \bmod M])\}_{(x,y)\in X}. \tag{1.24}$$

Then, the *set* V with the operations of addition, product, and multiplication by an element of the field of real numbers introduced on it is *DIA1R over the field of real numbers*.

The proof is based on verifying the properties of the classical algebra of the set V with the operations of addition, product, and multiplication by an element of the field of real numbers introduced on it (by Theorem 1.1).

1.4.1.5 *DIA1R1.5 over gray-scale images with matrix operations of addition and product*

We consider **DIA1R1.5** over gray-scale images with matrix image processing operations. Existence of this algebra is confirmed by the following theorem.

Theorem 1.6. *Suppose* R *is the field of real numbers*; *the elements of the set* U_4 *are the images of the following type* $I = \{((x,y), M[x,y])\}_{(x,y)\in X}; X = \{1,\dots,n;1,\dots,n\}$ *is the set of pixels the images are given at*; $I_1 = \{((x,y), M_1[x,y])\}_{(x,y)\in X}, I_2 = \{((x,y), M_2[x,y])\}_{(x,y)\in X}$ *are any two images from the set* U_4; *the quadratic matrices* $M, M_1,$ *and* M_2 *describe the entire set of values of the images* $I, I_1,$ *and* I_2, *respectively, given on the set* X. *We introduce*

- *the operation of addition of two images* I_1 *and* I_2

$$I_1 + I_2 = \{((x,y), (M_1 + M_2)[x,y])\}_{(x,y)\in X}, \tag{1.25}$$

- *the operation of product of two images* I_1 *and* I_2

$$I_1 \cdot I_2 = \{((x,y),(M_1 \cdot M_2)[x,y])\}_{(x,y)\in X}, \tag{1.26}$$

- *the operation of multiplication of the image* I *by the element of the field of real numbers* $\alpha \in R$

$$\alpha \cdot I = \{((x,y),(\alpha \cdot M)[x,y])\}_{(x,y)\in X}. \tag{1.27}$$

Then, the *set* U_4 with the operations of addition, product, and multiplication by an element of the field of real numbers introduced on it is *DIA1R over the field of real numbers.*

The proof is based on verifying the properties of the classical algebra of the set U_4 with the operations of addition, product, and multiplication by an element of the field of real numbers introduced on it (by Theorem 1.1).

The physical sense of operation (1.25): The operation is matched to matrix addition of two images.

The physical sense of operation (1.26): The operation is matched to matrix product of two images.

The physical sense of operation (1.27): The operation of multiplication of the image given by a matrix by a real number.

Proposition 1.10. *Introduced operations* (1.25) *and* (1.26) *of matrix addition and product of two images are* ***physically interpretable in the weak sense****.*

Proposition 1.11. *Introduced operation* (1.27) *of multiplication of the image by an element of the field of real numbers is* ***physically interpretable in the strong sense.***

1.4.1.6 *DIA1.6 over binary images*

We consider **DIA1.6** over binary images with logical operations of image processing. This algebra is not a classical algebra with one ring over a field yet it is an algebra of logic. Existence of this algebra is confirmed by the following theorem.

Theorem 1.7. *Suppose* $I_b = \{((x,y),bool(x,y))\}_{(x,y)\in X}$, *where* $bool(x,y) \in \{0,1\}$ *are the elements of the set* B, *and* $I_{b1} = \{((x,y)$

https://avxhm.se/blogs/hill0

$bool_1(x,y))\}_{(x,y)\in X}$, $I_{b2} = \{((x,y), bool_2(x,y))\}_{(x,y)\in X}$, *where* $bool_1(x,y), bool_2(x,y) \in \{0,1\}$ *are any elements of the set* B. *We introduce*

- *the operation of addition of two images* I_1 *and* I_2

$$I_{b1} + I_{b2} = \{((x,y), (\mathrm{bool}_1(x,y) + \mathrm{bool}_2(x,y)) \bmod 2)\}_{(x,y)\in X}, \tag{1.28}$$

- *the operation of product of two images* I_1 *and* I_2

$$I_{b1} \cdot I_{b2} = \{((x,y), \mathrm{bool}_1(x,y) \text{ AND } \mathrm{bool}_2(x,y))\}_{(x,y)\in X}, \tag{1.29}$$

- *the operation of multiplication of the image* I *by any binary number* $\alpha \in \{0,1\}$

$$\alpha \cdot I_b = \{((x,y), \alpha \cdot \mathrm{bool}(x,y))\}_{(x,y)\in X}. \tag{1.30}$$

Then, the *set* B with the operations of addition, product, and multiplication by a binary number introduced on it is *DIA*.

The proof is based on verifying the properties of algebra of logic of the set B with the operations of addition, product, and multiplication by a binary number introduced on it.

The physical sense of operation (1.28): The operation is matched to the complement of two binary images at each point (two masks are interlaid and only those areas are left that do not coincide on these masks).

The physical sense of operation (1.29): The operation is matched to the intersection of two binary images at each point (two masks are interlaid and only their intersection areas are left).

The physical sense of operation (1.30): The operation is matched to multiplication of the binary mask of the color image by 0 or by 1.

Proposition 1.12. *Introduced operations* (1.28) *and* (1.29) *of addition and product of the binary masks of two color images and operation* (1.30) *of multiplication of the binary mask of the color image by an element of the field of real numbers are* ***physically interpretable in the strong sense.***

1.4.2 *Descriptive image algebras with one ring over image transformations*

1.4.2.1 *DIA1R2.1 over procedural image transformations from the set of binary operations of SIA*

We again turn to operations of Ritter's SIA [102] for the sake of visual effect. Note that, in DAp, operations of Ritter's standard IA are considered as procedural transformations.

We consider **DIA1R2.1** over procedural image transformations from the set of binary operations of SIA. Existence of this algebra is confirmed by the following theorem.

Theorem 1.8. *Suppose the images are given on the fixed set X with the range of values X: $A = \{(x, a(x)), x \in X, a(x) \in X\}$. Suppose $B = \{(x, b(x)), x \in X, b(x) \in X\}$, $C = \{(x, c(x)), x \in X, c(x) \in X\}$. Suppose $\alpha \in R$.*

Suppose the following operations over images are introduced:

$$A + B = \{(x, c(x)) : c(x) = a(x) + b(x), x \in X\}, \tag{1.31}$$

$$A \cdot B = \{(x, c(x)) : c(x) = a(x) \cdot b(x), x \in X\}, \tag{1.32}$$

$$A \vee B = \{(x, c(x)) : c(x) = a(x) \vee b(x), x \in X\}, \tag{1.33}$$

$$A \wedge B = \{(x, c(x)) : c(x) = a(x) \wedge b(x), x \in X\}, \tag{1.34}$$

$$\frac{A}{B} = \left\{ (x, c(x)) : c(x) = \frac{a(x)}{b(x)} \quad \text{if } b(x) \neq 0, \right.$$
$$\left. \text{otherwise } c(x) = 0, x \in X \right\}, \tag{1.35}$$

$$A^B = \{(x, c(x)) \colon c(x) = a(x)^{b(x)} \quad \text{if } a(x) > 0,$$
$$\text{otherwise } c(x) = 0, x \in X\}, \tag{1.36}$$

$$A - B = \{(x, c(x)) \colon c(x) = a(x) - b(x), x \in X\}. \tag{1.37}$$

Suppose $r_1, r_2, \ldots \in \{+, *, \vee, \wedge, -, \backslash, A^B \cdots\}$ are standard binary operations over two images — the elements of the set R. $r(A, B)$ is the image after the operation r is applied to the images A and B.

On the set of the chosen operands, we give the following operations:

$$(r_1 \oplus r_2)(A, B) = r_1(A, B) + r_2(A, B), \tag{1.38}$$

$$(r_1 \otimes r_2)(A, B) = r_1(r_2(A, B), r_2(A, B)), \tag{1.39}$$

$$(\alpha r)(A, B) = \alpha r(A, B). \tag{1.40}$$

Then, the *set* R with the operations of addition, product, and multiplication by an element of the field of real numbers introduced on it is *DIA1R over the field of real numbers.*

The proof is based on verifying the properties of the classical algebra of the set R with the operations of addition, product, and multiplication by an element of the field of real numbers introduced on it (by Theorem 1.1).

The physical sense of operation (1.38): The operation is matched to the operations r_1 and r_2 applied successively and followed by addition of the images constructed as a result of the operations r_1 and r_2 applied to the original images.

The physical sense of operation (1.39): The operation is matched to the first binary transformation applied to the result of the second binary image transformation taken twice and applied to the original images.

The physical sense of operation (1.40): The operation is matched to the multiplication of the result of application of the binary image transformation by an element of the field of real numbers.

Proposition 1.13. *Operation* (1.38) *of the operations* r_1 *and* r_2 *applied successively and followed by addition of the images constructed as a result of the operations* r_1 *and* r_2 *applied to original images* ***is not physically and visually interpretable.***

Proof.

1. By Proposition 1.2, the operation is physically non-interpretable in the context of image analysis and recognition if images, image representations, and image fragments are not its operands.

2. By Corollary from Proposition 1.1, if the operation is physically non-interpretable, it is also visually non-interpretable.

Q.E.D.

An example of using this operation is that this operation can be applied to combine operations over images.

Proposition 1.14. *Operation* (1.39) *of the first binary transformation* r_1 *applied to the result of the second binary image transformation* r_2 *taken twice and applied to the original images* ***is not physically and visually interpretable****.*

The proof is similar to the proof of the previous proposition.

An example of using this operation is that this operation can be applied to combine operations over images.

Proposition 1.15. *Operation* (1.40) *of multiplication of the result of application of the binary image transformation by an element of the field of real numbers* ***is not physically and visually interpretable.***

The proof is similar to the proof of Proposition 1.13.

An example of using this operation is that this operation can be applied to correct the brightness and color of an image after the operation r is applied to it.

1.4.2.2 *DIA1R2.2 over procedural transformations of color images into gray-scale ones*

Suppose now the operations $f(U \to V) \in F$ of converting the elements of the set of color images described in *Section* 1.4.1.3 *DIA1R1.3 over color images* into the elements of the set of gray-scale images described in *Section* 1.4.1.4. *DIA1R1.4 over gray-scale images*.

We consider **DIA1R2.2** over procedural transformations of color images into gray-scale ones. Existence of this algebra is confirmed by the following theorem.

Theorem 1.9. *Suppose* $I = \{((x,y),(r(x,y),g(x,y),b(x,y))\}_{(x,y)\in X}$, *where the functions* $r(x,y), g(x,y), b(x,y)$ *take integer values from the set* $[0 \cdots M-1]$. *Suppose* $J = \{((x,y),gray(x,y))\}_{(x,y)\in X}$,

$J_1 = \{((x,y), gray_1(x,y))\}_{(x,y)\in X}$, $J_2 = \{((x,y), gray_2(x,y))\}_{(x,y)\in X}$, *where the functions* $gray(x,y), gray_1(x,y), gray_2(x,y)$ *take integer values from the set* $[0 \cdots M-1]$. $M = 256$ *is the value of the maximal brightness of color components*; X *is the set of pixels images are given at*; *and* $f_1(U \to V) \in F, f_2(U \to V) \in F$ *are the operations of converting the elements of the set* U *into the elements of the set* V. *Suppose* $f_1(I) = J_1, f_2(I) = J_2, f(I) = J$. *Suppose* $\alpha \in R$.

On the set of the chosen operands, we give the following operations:

$$f_1(I) + f_2(I) = J_1 + J_2, \tag{1.41}$$

$$f_1(I) \cdot f_2(I) = J_1 \cdot J_2, \tag{1.42}$$

$$\alpha \cdot f(I) = \alpha \cdot J. \tag{1.43}$$

Then, the *set* F with the operations of addition, product, and multiplication by an element of the field of real numbers introduced on it is *DIA1R over the field of real numbers*.

The proof is based on verifying the properties of the classical algebra of the set F with the operations of addition, product, and multiplication by an element of the field of real numbers introduced on it (by Theorem 1.1).

The physical sense of operation (1.41): The operation is matched to addition of the results of application of two operations of converting the original color image into gray-scale images. Note that addition of the resulting gray-scale images is given in DIA1R1.4.

The physical sense of operation (1.42): The operation is matched to product of the results of application of two operations of converting the original color image into gray-scale images. Note that product of two gray-scale images is given in DIA1R1.4.

The physical sense of operation (1.43): The operation is matched to multiplication of the result of application of the operation of converting the original color image into a gray-scale image by a real number. Note that multiplication of a gray-scale image by a real number is given in DIA1R1.4.

Proposition 1.16. *Operations* (1.41), (1.42), *and* (1.43) *of addition, product, and multiplication by a real number, respectively, of the results of applying the operations of converting the original color*

image into gray-scale images ***are not physically and visually interpretable.***

The proof is similar to the proof of Proposition 1.15.

Similarly, one can introduce DIA over procedural transformations of gray-scale images into binary ones.

1.4.2.3 *DIA1R2.3 over procedural transformations of obtaining fragments of gray-scale images represented in the matrix form*

We remind that the set U_4 consists of the images of the following form $I = \{((x,y), M[x,y])\}_{(x,y)\in X}$. $X = \{1,\dots,n;1,\dots,n\}$ is the set of pixels images are given at.

We introduce the operation of obtaining the image fragment $\mathrm{fr}[l,u,m]\,(I) = J$, where (l,u) is the coordinate of the upper left angle of the fragment and m is the size of the fragment in pixels, such that $(l,u) \in X, (l+m, u+m) \in X$. $J = \{((x,y), M[x,y])\}_{(x,y)\in(l\dots l+m,\, u\dots u+m)\subset X}$.

We consider **DIA1R2.3** over procedural transformations of obtaining fragments of gray-scale images represented in the matrix form. Existence of this algebra is confirmed by the following theorem.

Theorem 1.10. *Suppose R is the field of real numbers; $X = \{1,\dots,n;1,\dots,n\}$ is the set of pixels the images of the form $I = \{((x,y), M[x,y])\}_{(x,y)\in X}$; $fr[l_1,u_1,m_1]$, $fr[l_2,u_2,m_2]$ are given at, where $(l_1 \dots l_1+m_1, u_1 \dots u_1+m_1) \subset X, (l_2 \dots l_2+m_2, u_2 \dots u_2+m_2) \subset X$ are the operations of singling out the fragments on the image from the set Fr.*

We introduce

- *the operation of addition of two operations of singling out fragments on the image fr_1 and fr_2*

$$\begin{aligned} &\mathrm{fr}[l_1,u_1,m_1] + \mathrm{fr}[l_2,u_2,m_2] = \{((x,y), M[x,y])\}, \\ &\quad \text{where} \quad (x,y) \in (l_1 \dots l_1+m_1, u_1 \dots, u_1+m_1) \\ &\quad \cup (l_2 \dots l_2+m_2, u_2 \dots u_2+m_2), \end{aligned} \tag{1.44}$$

- *the operation of product of two operations of singling out fragments on the image I_1 and I_2*

$$\begin{aligned}\mathrm{fr}[l_1,u_1,m_1]\cdot\mathrm{fr}[l_2,u_2,m_2]&=\{((x,y),M[x,y])\},\\ \text{where}\quad (x,y)&\in(l_1\dots l_1+m_1,u_1\dots,u_1+m_1)\\ \cap(l_2\dots l_2+m_2,u_2\dots u_2+m_2),&\end{aligned}\tag{1.45}$$

- *the operation of multiplication of the operation of singling out a fragment on the image I by the element of the field of real numbers $\alpha\in R$*

$$\begin{aligned}\alpha\cdot\mathrm{fr}[l,u,m]&=\{((x,y),M)[x,y]\}.\\ \text{where}\quad (x,y)&\in(l\dots l+[\alpha\cdot m],u\dots u+[\alpha\cdot m]).\end{aligned}\tag{1.46}$$

Then, the *set Fr* with the operations of addition, product, and multiplication by an element of the field of real numbers introduced on it is *DIA1R over the field of real numbers*.

The proof is based on verifying the properties of the classical algebra of the set *Fr* with the operations of addition, product, and multiplication by an element of the field of real numbers introduced on it (by Theorem 1.1).

The physical sense of operation (1.44): The operation is matched to the union of two image fragments.

The physical sense of operation (1.45): The operation is matched to intersection of two image fragments.

The physical sense of operation (1.46): The operation is matched to scaling of an image fragment.

Proposition 1.17. *Operations* (1.44), (1.45), *and* (1.46) *of addition, product, and multiplication by a real number of the operations of singling out the fragments of gray-scale images represented in the matrix form* ***are physically interpretable in the strong sense.***

1.4.2.4 *DIA1R2.4 over procedural transformations of filtering of gray-scale images*

We consider the set U_4 of images of the following form $I=\{((x,y),M[x,y])\}_{(x,y)\in X}$.

We introduce the operation of filtering of the image $\mathrm{ftr}[A](I) = J$, where A is the matrix of dimension $\boldsymbol{l \times l}$, $\boldsymbol{l \in N, l < n}$ that gives the kernel of the filter applied to each point of the original image I to obtain a new filtered image J.

We consider **DIA1R2.4** over procedural transformations of filtering of gray-scale images represented in the matrix form. Existence of this algebra is confirmed by the following theorem.

Theorem 1.11. *Suppose R is the field of real numbers; $X = \{1, \ldots, n; 1, \ldots, n\}$ is the set of pixels the images of the form $I = \{((x, y), M[x, y])\}_{(x,y)\in X}$; $ftr[A_1]$, $ftr[A_2]$ are given at, where A_1 and A_2 are the kernels of the filters — the operation of filtering of the image from the set Ftr.*

We introduce

- *the operation of addition of two operations of filtering of the image I $ftr[A_1]$* and *$ftr[A_2]$*

$$\mathrm{ftr}[A_1] + \mathrm{ftr}[A_2] = J_1 + J_2, \tag{1.47}$$

- *the operation of product of two operations of filtering of the image I $ftr[A_1]$* and *$ftr[A_2]$*

$$\mathrm{ftr}[A_1]^* \mathrm{ftr}[A_2] = J_1^* J_2, \tag{1.48}$$

- *the operation of multiplication of the operation of filtering of the image I by the element of the field of real numbers $\alpha \in R$*

$$\alpha \cdot \mathrm{ftr}[A] = \alpha \cdot J. \tag{1.49}$$

Then, the *set Ftr* with the operations of addition, product, and multiplication by an element of the field of real numbers introduced on it is *DIA1R over the field of real numbers*.

The proof is based on verifying the properties of the classical algebra of the set *Ftr* with the operations of addition, product, and multiplication by an element of the field of real numbers introduced on it (by Theorem 1.1).

The physical sense of operation (1.47): The operation is matched to successive application of the filtering operations to the original image and addition of the transformed images. Note that addition of two gray-scale images is given in DIA1R1.5.

The physical sense of operation (1.48): The operation is matched to successive application of the filtering operations to the original image and product of the transformed images. Note that product of two gray-scale images is given in DIA1R1.5.

The physical sense of operation (1.49): The operation is matched to successive application of the filtering operation to the original image and multiplication of the transformed image by an element of the field. Note that multiplication of a gray-scale image by an element of the field is given in DIA1R1.5.

Proposition 1.18. *Operations* (1.47), (1.48), *and* (1.49) *of addition, product, and multiplication by a real number of the operations of filtering of gray-scale images represented in the matrix form* ***are physically interpretable in the weak sense.***

1.4.2.5 *DIA2.5 over procedural transformations of binary images*

We remind that $I_b = \{((x,y), bool(x,y))\}_{(x,y)\in X}$, where $bool(x,y) \in \{0,1\}$ are the elements of the set B. Suppose the transformations $b(I_b) = J_b$ are introduced and transform binary images into binary ones from the set B represented as DIA1.6.

We consider **DIA2.5** over procedural transformations of binary images. Existence of this algebra of logic is confirmed by the following theorem.

Theorem 1.12. *Suppose* $I_b = \{((x,y), bool(x,y))\}_{(x,y)\in X}$, *where* $bool(x,y) \in \{0,1\}$ *are the elements of the set B*; *X is the set of pixels the images are given at*; *and* $b_1(B \to B) \in Fb, b_2(B \to B) \in Fb$ *are the operations of transformations of binary images of the set B. Suppose* $b_1(I) = J_{b1}, b_2(I) = J_{b2}, b(I) = J_b, J_{b1}, J_{b2}, J_b \in B$. *Suppose* $\alpha \in \{0,1\}$.

On the set of the chosen operands, the following operations are given:

$$b_1(I) + b_2(I) = J_{b1} + J_{b2}, \tag{1.50}$$

$$b_1(I) \cdot b_2(I) = J_{b1} \cdot J_{b2}, \tag{1.51}$$

$$\alpha \cdot b(I) = \alpha \cdot J_b. \tag{1.52}$$

Then, the *set Fb* with the operations of addition, product, and multiplication by a binary number introduced on it is *DIA over binary numbers*.

The proof is based on checking the algebra of logic of the set *Fb* with the operations of addition, product, and multiplication by a binary number introduced on it.

The physical sense of operation (1.50): The operation is matched to addition of the results of applying two operations of transformation of binary images. Note that addition of the resulting binary images is given in DIA1.6.

The physical sense of operation (1.51): The operation is matched to product of the results of applying two operations of transformation of binary images. Note that product of the resulting binary images is given in DIA1.6.

The physical sense of operation (1.52): The operation is matched to multiplication of the result of applying the operation of transformation of a binary image by a number from the set $\{0, 1\}$. Note that multiplication of a binary image by the number $\alpha \in \{0, 1\}$ is given in DIA1.6.

Proposition 1.19. *Operations* (1.50), (1.51), *and* (1.52) *of addition, product, and multiplication by a binary number, respectively, of transformations of binary images* ***are physically interpretable in the weak sense.***

1.4.2.6 *DIA1R2.6 over parametric transformations*

We remind that the elements of the set V are $J = \{((x, y), gray(x, y))\}_{(x,y)\in X}$, where the function $gray(x, y)$ takes integer values from the set $[0 \cdots M - 1]$, $M = 256$ is the number of hues, and X is the set of pixels images are given at. The set V is described by DIA1R1.3.

We consider **DIA1R2.6** over parametric transformations. Existence of this algebra is confirmed by the following theorem.

Theorem 1.13. *Suppose the operations* $g(V \to P_1)$ *of calculating the values of features of gray-scale images are the elements of the set* G, P_1 *is the set of parametric image models described in the following,* $J = \{((x, y), gray(x, y))\}_{(x,y)\in X}$, *where the function* $gray(x, y)$ *takes*

integer values from the set $[0 \cdots M-1]$, $M = 256$ *is the number of hues of brightness*, X *is the set of pixels images are given at*, $g_1(J) = p_1, g_2(J) = p_2, g(J) = p$, *and suppose* $\alpha \in R$.

We introduce the following operations over the given operands:

$$g_1(I) + g_2(I) = p_1 + p_2, \tag{1.53}$$

$$g_1(I) \cdot g_2(I) = p1 \cdot p2, \tag{1.54}$$

$$\alpha \cdot g(I) = \alpha \cdot p. \tag{1.55}$$

Then, the *set* G with the operations of addition, product, and multiplication by an element of the field of real numbers introduced on it is *DIA1R over the field of real numbers*.

The proof is based on verifying the properties of the classical algebra of the set G with the operations of addition, product, and multiplication by an element of the field of real numbers introduced on it (by Theorem 1.1).

The physical sense of operation (1.53): The operation is matched to the operation of addition of two parametric image models p_1 and p_2 obtained after the operations of calculation of the values of features of a gray-scale image g_1 and g_2 are applied.

The physical sense of operation (1.54): The operation is matched to the operation of product of two parametric image models p_1 and p_2 obtained after the operations of calculation of the values of features of a gray-scale image g_1 and g_2 are applied.

The physical sense of operation (1.55): The operation is matched to the operation of multiplication by an element of the field of real numbers of the parametric image model p obtained after the operation of calculation of the values of features of a gray-scale image g is applied.

Proposition 1.20. *Operations* (1.53), (1.54), *and* (1.56) *of addition, product, and multiplication by a real number, respectively, of the results of applying the operations of calculating the values of features of a gray-scale image* ***are not physically and visually interpretable.***

The proof is similar to the proof of Proposition 1.19.

1.4.3 *Descriptive image algebras with one ring over image representations/models*

1.4.3.1 *DIA1R3.1 over various parametric representations/models*

We consider **DIA1R3.1** over various parametric representations/models. Existence of this algebra is confirmed by the following theorem.

Suppose P_1 is the set of parametric image models such that any element $\overline{p}$ of the set P_1 is given by the vector of the following form $\overline{p} = (p^1, p^2, \ldots, p^n)$, where $n = 1, 2, \ldots$ is the number of features describing the parametric image model, $p^i = (p^i_x, p^i_y), i = 1, \ldots, n$ are the elements of the vector, $p^i_x = 1, 2, \ldots$ is the unique number of the feature in the feature space that gives the set P_1, p^i_y is the value of the feature from the set of real numbers, and $p^i_x < p^j_x$ if $i < j$; $p^i_{1y} = p^j_{2y}$ if $p^i_{1x} = p^j_{2x}$.

Theorem 1.13. *Suppose* $\overline{p_1} = (p^1_1, p^2_1, \ldots, p^{n_1}_1)$, $\overline{p_2} = (p^1_2, p^2_2, \ldots, p^{n_2}_2)$ *are the elements of the set* P_1; *suppose* $\alpha \in R$.

We introduce the following operations over the parametric image models:

$$\begin{aligned}
\overline{p_1} + \overline{p_2} &= (p^1_1, p^2_1, \ldots, p^{n_1}_1) + (p^1_2, p^2_2, \ldots, p^{n_2}_2) \\
&= (p^1_3, p^2_3, \ldots, p^{n_3}_3), \\
p^i_{3x} &\subset [p^1_{1x}, \ldots, p^{n_1}_{1x}] \cap [p^1_{2x}, \ldots, p^{n_2}_{2x}], \\
n_3 &= \|[p^1_{1x}, \ldots, p^{n_1}_{1x}] \cap [p^1_{2x}, \ldots, p^{n_2}_{2x}]\|, \\
p^i_{3y} &= p^j_{1y} \quad \text{if } p^i_{3x} = p^j_{1x}, \\
p^i_{3y} &= p^j_{2y} \quad \text{if } p^i_{3x} = p^j_{2x}, \qquad (1.56) \\
\overline{p_1} \cdot \overline{p_2} &= (p^1_1, p^2_1, \ldots, p^{n_1}_1) \cdot (p^1_2, p^2_2, \ldots, p^{n_2}_2) \\
&= (p^1_4, p^2_4, \ldots, p^{n_4}_4), \qquad (1.57)
\end{aligned}$$

$n_4 < n_3, (p^1_4, p^2_4, \ldots, p^{n_4}_4)$ *are the essential features,*

$$\begin{aligned}
\alpha \cdot \overline{p} &= (\alpha \cdot p^1, \ldots, p^n) = ((p^1_x, \alpha \cdot p^1_y), \ldots, (p^n_x, \alpha \cdot p^n_y)), \\
\alpha \cdot \overline{p} &= \alpha \cdot (p^1, p^2, \ldots, p^n) = ((p^1_x, \alpha \cdot p^1_y), \ldots, (p^n_x, \alpha \cdot p^n_y)). \qquad (1.58)
\end{aligned}$$

Then, the *set* P_1 with the operations of addition, product, and multiplication by an element of the field of real numbers introduced on it is *DIA1R over the field of real numbers*.

The proof is based on verifying the properties of the classical algebra of the set P_1 with the operations of addition, product, and multiplication by an element of the field of real numbers introduced on it (by Theorem 1.1).

The physical sense of operation (1.56): The operation is matched to the operation of addition of two parametric image models, which leads to joining the parametric models so that they include all unique features from both models.

The physical sense of operation (1.57): The operation is matched to the operation of product of two parametric image models, which leads to singling out the most informative features describing the object after the original parametric models are analyzed.

The physical sense of operation (1.58): The operation is matched to the operation of multiplication of the parametric image model by a real number so that the values of features are multiplied by a real number as a result.

Proposition 1.21. *Operations* (1.56), (1.57), *and* (1.58) *of addition, product, and multiplication by a real number, respectively, of the parametric models* ***are physically interpretable in the weak sense.***

The proof is similar to the proof of the previous propositions.

1.4.3.2 *DIA1R3.2 over parametric representations/models*

We consider **DIA1R3.2** over parametric representations/models. Existence of this algebra is confirmed by the following theorem.

Theorem 1.14. *Suppose* P_2 *is the set of parametric image models such that any element* $\overline{p}$ *of the set* P_2 *is given by the vector of the following form* $\overline{p} = (p^1, p^2, \ldots, p^n)$, *where* $n = 1, 2, \ldots$ *is the number of features describing the parametric image model,* n *is a fixed number of features, and* p^i *are the elements of the vector* $(i = 1, \ldots, n)$ *from*

the set of real numbers. Suppose $\overline{p}_1, \overline{p}_2$ *are any elements of the set* P_2; *suppose* $\alpha \in R$.

We introduce the following operations over parametric image models:

$$\begin{aligned}\overline{p}_1 + \overline{p}_2 &= (p_1^1, p_1^2, \ldots, p_1^n) + (p_2^1, p_2^2, \ldots, p_2^n) \\ &= (p_1^1 + p_2^1, p_1^2 + p_2^2, \ldots, p_1^n + p_2^n), \end{aligned} \tag{1.59}$$

$$\begin{aligned}\overline{p}_1 \cdot \overline{p}_2 &= (p_1^1, p_1^2, \ldots, p_1^n) \cdot (p_2^1, p_2^2, \ldots, p_2^n) \\ &= (p_1^1 \cdot p_2^1, p_1^2 \cdot p_2^2, \ldots, p_1^n \cdot p_2^n), \end{aligned} \tag{1.60}$$

$$\alpha \cdot \overline{p} = (\alpha \cdot p^1, \alpha \cdot p^2, \ldots, \alpha \cdot p^n). \tag{1.61}$$

Then, the *set* P_2 with the operations of addition, product, and multiplication by an element of the field of real numbers introduced on it is *DIA1R over the field of real numbers*.

The proof is based on verifying the properties of the classical algebra of the set P_2 with the operations of addition, product, and multiplication by an element of the field of real numbers introduced on it (by Theorem 1.1).

The physical sense of operation (1.59): The operation is matched to the operation of addition of two parametric image models, which leads to the values of identical features in the parametric models being summed up.

The physical sense of operation (1.60): The operation is matched to the operation of product of two parametric image models, which leads to the values of identical features in the parametric models being multiplied.

The physical sense of operation (1.61): The operation is matched to the operation of multiplication of each feature of the parametric image model by a real number.

Proposition 1.22. *Operations* (1.59), (1.60), *and* (1.61) *of addition, product, and multiplication by a real number, respectively, of the parametric models* ***are physically interpretable in the weak sense.***

1.5 Algebraic Model of Solving the Problem of Automatization of Detection of Abnormal Changes in Morphometrical Characteristics of Human's Fundus

We give the algebraic model of solving the problem of OCT-A image analysis for automatization of detection of abnormal changes in morphometrical characteristics of human's fundus (for a detailed description of the problem, see [30]). This algebraic model generates a set of specialized DAS when one refines the particular transformations and representations from the classes of specialized DIA and DIM.

The sign $\rightarrow$ in (1.62)–(1.73) designates that the transformation of the respective algebra on its left is applied to the operand on its right. I_2 is the original image.

We can write the algebraic model as follows:

Step 1: Within the algebra of color images (see Section 1.4.1.3), the image is manipulated with

$$\text{DIA1R1.3} \rightarrow I_2 = \text{DIA1R1.3}(I_2). \tag{1.62}$$

Step 2: Within the algebra of procedural transformations (see Section 1.4.2.2), the color image is transformed into a gray-scale one:

$$\text{DIA1R2.2} \rightarrow \text{DIA1R1.3}(I_2) = \text{DIA1R2.2}(\text{DIA1R1.3}(I_2)). \tag{1.63}$$

Step 3: Within the algebra of gray-scale images (see Section 1.4.2.2), the image fragments are manipulated with

$$\begin{aligned} &\text{DIA1R1.4} \rightarrow \text{DIA1R2.2}(\text{DIA1R1.3}(I_2)) \\ &\quad = \text{DIA1R1.4}(\text{DIA1R2.2}(\text{DIA1R1.3}(I_2))). \end{aligned} \tag{1.64}$$

Step 4: Within the algebra over matrices (see Section 1.4.1.5), the gray-scale image is transformed into the matrix form for the following transformations to be further applied:

$$\begin{aligned}\text{DIA1R1.5} \rightarrow\ & \text{DIA1R1.4}(\text{DIA1R2.2}(\text{DIA1R1.3}(I_2))) \\ & = \text{DIA1R1.5}(\text{DIA1R1.4}(\text{DIA1R2.2}(\text{DIA1R1.3}(I_2)))). \quad (1.65)\end{aligned}$$

Step 5: Within the algebra of image fragments (see Section 1.4.2.3), necessary fragments are singled out on the image; for instance, the image model $M_{T1}(I_2)$ is constructed (see Section 1.2.3):

$$\begin{aligned}\text{DIA1R2.3} \rightarrow\ & \text{DIA1R1.5}(\text{DIA1R1.4}(\text{DIA1R2.2}(\text{DIA1R1.3}(I_2)))) \\ & = \text{DIA1R2.3}(\text{DIA1R1.5}(\text{DIA1R1.4}(\text{DIA1R2.2}(\text{DIA1R1.3}(I_2))))) \\ & = M_{T1}(I_2). \quad (1.66)\end{aligned}$$

Step 6: Within the algebra of gray-scale images (see Section 1.4.1.5), the image fragments represented in the matrix form are manipulated with

$$\text{DIA1R1.5} \rightarrow\ M_{T1}(I_2) = \text{DIA1R1.5}(M_{T1}(I_2)). \quad (1.67)$$

Step 7: Within the algebra of filtering transformations (see Section 1.4.2.4), the filtering operations are chosen and applied:

$$\begin{aligned}\text{DIA1R2.4} \rightarrow\ & \text{DIA1R1.5}(M_{T1}(I_2)) \\ & = \text{DIA1R2.4}(\text{DIA1R1.5}(M_{T1}(I_2))). \quad (1.68)\end{aligned}$$

Step 8: Within the algebra of transformations of gray-scale images into binary images (see Section 1.4.2.2), the gray-scale image is transformed into the binary image:

$$\begin{aligned}\text{DIA2.2.1} \rightarrow\ & \text{DIA1R2.4}(\text{DIA1R1.5}(M_{T1}(I_2))) \\ & = \text{DIA2.2.1}(\text{DIA1R2.4}(\text{DIA1R1.5}(M_{T1}(I_2)))) \\ & = M_{T2}(I_2). \quad (1.69)\end{aligned}$$

Step 9: Within the algebra of binary images (see Section 1.4.1.6), the binary representation of an image is manipulated with

$$\text{DIA1.6} \rightarrow M_{T2}(I_2) = \text{DIA1.6}(M_{T2}(I_2)) \quad (1.70)$$

Step 10: Within the algebra of transformations of binary images (see Section 1.4.2.5), binary transformations of binary image representations are chosen and applied; for instance, the following

procedural models can be obtained $M_{T3}(I_2))$, $M_{T4}(I_2)$, $M_{T5}(I_2))$ (see Section 1.2.3):

$$\begin{aligned}\text{DIA2.5} \to \text{DIA1.6}(M_{T2}(I_2)) &= \text{DIA2.5}(\text{DIA1.6}(M_{T2}(I_2))) \\ &= \{M_{T3}(I_2), M_{T4}(I_2), M_{T5}(I_2)\}.\end{aligned} \tag{1.71}$$

Step 11: Within the algebra of parametric transformations (see Section 1.4.2.6), parametric transformations are chosen and applied to calculate features by the constructed image models; for instance, the following parametric image models can be obtained $M_{P1}(I_2)$, $M_{P2}(I_2)$, $M_{P3}(I_2))$ (see Section 1.2.3):

$$\begin{aligned}&\text{DIA1R2.6} \to \{M_{T1}(I_2), M_{T2}(I_2), M_{T3}(I_2), M_{T4}(I_2), M_{T5}(I_2)\} \\ &= \text{DIA1R2.6}(\{M_{T1}(I_2), M_{T2}(I_2), M_{T3}(I_2), M_{T4}(I_2), M_{T5}(I_2)\}) \\ &= \{M_{P1}(I_2), M_{P2}(I_2), M_{P3}(I_2)\}.\end{aligned} \tag{1.72}$$

Step 12: Within the algebra over various parametric representations (see Section 1.4.3.1), the essential features for all classes of the recognition problem are chosen:

$$\begin{aligned}&\text{DIA1R3.1} \to \{M_{P1}(I_2), M_{P2}(I_2), M_{P3}(I_2)\} \\ &= \text{DIA1R3.1}(\{M_{P1}(I_2), M_{P2}(I_2), M_{P3}(I_2)\}).\end{aligned} \tag{1.73}$$

Step 13: Within the algebra over parametric representations (see Section 1.4.3.2), the chosen features are corrected:

$$\begin{aligned}&\text{DIA1R3.2} \to \text{DIA1R3.1}(\{M_{P1}(I_2), M_{P2}(I_2), M_{P3}(I_2)\}) \\ &= \text{DIA1R3.2}(\text{DIA1R3.1}(\{M_{P1}(I_2), M_{P2}(I_2), M_{P3}(I_2)\})).\end{aligned} \tag{1.74}$$

One can apply various transformations of image representations within each class of algebras. In the previous sections of this chapter, we introduced the principal theoretical aspects needed to understand the proposed ways and tools of DAS description. We give a more detailed description of the model in our future works. Algebraic models (1.62)–(1.74) of solving the problem generate a set of image analysis DAS.

In [30], an example of DAS for solving the problem of analyzing OCT-angiograms to automate the detection of abnormal changes in morphometrical fundus characteristics is given.

1.6 Conclusions

The fundamental problem of the carried-out research is automatization of extracting information from images that is necessary for intellectual decisions to be made.

In the future, we suggest defining, constructing, and studying the generalized informational structure that allows both refining the recognition methodology, including the mathematical apparatus, and simulating structural properties of recognition processes, such as them being multilevel, multialgorithmic, and multimodel. The strategical goal is the model of a universal computer for image analysis — the standard informational structure used to represent algorithms of analysis and recognition of two-dimensional information and to represent procedures of constructing dual image representations in the form of DAS. Automatization and regularization of generation of such DAS will allow us to generalize and substantiate the known heuristic recognition algorithms, perform their comparative analysis, and evaluate their mathematical properties and applicative value.

The crucial importance of the results of this research for the development of the mathematical theory of image analysis and their scientific novelty is associated with problem statements and the development of methods of simulating automatization processes of image analysis when ill-formalized image representations are used as the input data and include the spatial data themselves, viz. images and their fragments, image models, not completely formalized representations, and subsets of combinations of the above-mentioned representations.

References

[1] Batcher, K.E. (1980) Design of a massively parallel processor. *IEEE Transactions on Computers*, C-29(9), 836–840.

[2] Barrow, H.G., Ambler, A.P., and Burstall, R.M. (1972) Some techniques for recognizing structures in pictures. In Watanabe, S. (ed.) *The Proceedings of the International Conference on Frontiers of Pattern Recognition,* pp. 1–30.

[3] Beloozerov, V.N., Gurevich, I.B., Gurevich, N.G., Murashov, D.M., and Trusova, Y.O. (2003) Thesaurus for image analysis: Basic version. *Pattern Recognition and Image Analysis: Advances in Mathematical Theory and Applications*, 13(4), 556–569.

[4] Beloozerov, V.N., Gurevich, I.B., Murashov, D.M., and Trusova, Y.O. (2003) Construction and use of a thesaurus in image analysis and processing. *Pattern Recognition and Image Analysis: Advances in Mathematical Theory and Application*, 13(1), 67–69.

[5] Beloozerov, V.N., Gurevich, I.B., and Trusova, Y.O. (2005) Representation of the ontology of an image analysis domain for optimization of information retrieval. *Pattern Recognition and Image Analysis: Advances in Mathematical Theory and Applications*. MAIK "Nauka/Interperiodica" (Russia), 15(2), 358–360.

[6] Birkhoff, G. and Lipson, J.D. (1970) Heterogeneous algebras. *Journal of Combinatorial Theory*, 8, 115–133.

[7] Chernov, V.M. (2001) Clifford algebras are group algebras projections. In Bayro-Corrochano, E. and Sobczyk, G. (eds.) *Advances in Geometric Algebra with Applications in Science and Engineering*, pp. 467–482.

[8] Chu, Z., *et al.* (2016) Quantitative assessment of the retinal microvasculature using optical coherence tomography angiography. *Journal of Biomedical Optics*, 21(6), 066008.

[9] Colantonio, S., Martinelli, M., Salvetti, O., Gurevich, I.B., and Trusova, Y.O. (2008) Cell image analysis ontology. *Pattern Recognition and Image Analysis: Advances in Mathematical Theory and Applications*, 18(2), 332–341.

[10] Colantonio, S., Salvetti, O., Gurevich, I., and Trusova, Y. (2009) An ontological framework for media analysis and mining. *Pattern Recognition and Image Analysis: Advances in Mathematical Theory and Applications*, 19(2), 221–230.

[11] Davidson, J.L. (1993) Classification of lattice transformations in image processing. *Computer Vision, Graphics, and Image Processing: Image Understanding*, 57(3), 283–306.

[12] Dougherty, E.R. (1989) A homogeneous unification of image algebra. Part I: The homogenous algebra. *Imaging Science*, 33(4), 136–143.

[13] Dougherty, E.R. (1989) A homogeneous unification of image algebra. Part II: Unification of image algebra. *Image Science*, 33(4), 144–149.

[14] Dougherty, E.R. and Sinha, D. (1995) Computational gray-scale mathematical morphology on lattices (a comparator-based image algebra). Part 1: Architecture. *Real-Time Imaging*, 1, 69–85.

[15] Dougherty, E.R. and Sinha, D. (1995) Computational gray-scale mathematical morphology on lattices (a comparator-based image algebra). Part 2: Image operators. *Real-Time Imaging*, 1, 283–295.

[16] Duff, M.J.B., Watson, D.M., Fountain, T.J., and Shaw, G.K. (1973) A cellular logic array for image processing. *Pattern Recognition*, 5(3), 229–247.

[17] Evans, T.G. (1969) Descriptive pattern analysis techniques: Potentialities and problems. *The Proceedings of the International Conference on Methodologies of Pattern Recognition*, pp. 149–157.

[18] Furman, Y.A. (2007) Visualization of images in three-dimensional scenes: Tutorial (Yoshkar-Ola, Mari State Technical University) [in Russian].

[19] Furman, Y.A. (2009) Parallel recognition of different classes of patterns. *Pattern Recognition and Image Analysis*, 19(3), 380–393.

[20] Grenander, U. (1976, 1978, 1981) *Lectures in Pattern Theory* (Sprinder-Verlag, New York).

[21] Grenander, U. (1993) *General Pattern Theory. A Mathematical Study of Regular Structure* (Clarendon Press, Oxford).

[22] Grenander, U. (1996) *Elements of Pattern Theory* (The Johns Hopkins University Press, Baltimore).

[23] Grenander, U. and Miller, M. (2007) *Pattern Theory: From Representation to Inference* (Oxford University Press, New York).

[24] Grin, J., Kittler, J., Pudil, P., and Somol, P. (2001) Information analysis of multiple classifier fusion. *Proceedings of the 2nd International Workshop Multiple Classifier Systems*, pp. 168–177.

[25] Gurevitch, I.B. (1985) The algebraic approach to image recognition and analysis. *Mathematical Techniques of Pattern Recognition: Proceedings of All-Union Conference* (Dilidgan, May 16–21 1985). Yerevan, The Publishing House of the Academy of Sciences of Armenian SSR, pp. 55–57 (in Russian).

[26] Gurevitch, I.B. (1986) The definition of the class of algorithms based on two-dimensional data estimates calculation for image recognition problems. In Vasin, Y.G. (ed.) *The Techniques and Means for Graphic Information Processing: Interuniversity Collection of Papers*, pp. 47–66 (in Russian).

[27] Gurevitch, I.B. (1989) The descriptive framework for an image recognition problem. *Proceedings of the 6th Scandinavian Conference on Image Analysis, Pattern Recognition Society of Finland*, Vol. 1, pp. 220–227.

[28] Gourevitch, I.B. (1995) A descriptive method for image analysis based on the synthesis of an image model in the class of disjunctive normal forms. *Pattern Recognition and Image Analysis: Advances in Mathematical Theory and Applications*, 5(3), 356–363.

[29] Gurevich, I. (2005) The descriptive approach to image analysis. Current state and prospects, image analysis. In Kalviainen, H., Parkkinen, J., and Kaarna, A. (eds.) *14th Scandinavian Conference*, SCIA2005, Joensuu, Finland, June 2005, Proceedings, LNCS, Vol. 3540, pp. 214–223.

[30] Gurevich, I.B., Budzinskaya, M.V., Yashina, V.V., Nedzved, A.M., Tleubaev, A.T., Pavlov, V.G., and Petrachkov, D.V. (2021) A new method for automating the diagnostic analysis of human fundus images obtained using optical coherent tomography angiography. *Pattern Recognition and Image Analysis*, 31(3), 513–528.

[31] Gurevich, I.B. and Jernova, I.A. (2003) The joint use of image equivalents and image invariants in image recognition. *Pattern Recognition and Image Analysis: Advances in Mathematical Theory and Applications*, 13(4), 570–578.

[32] Gurevich, I.B., Khilkov, A.V., Koryabkina, I.V., Murashov, D.M., and Trusova, Y.O. (2006) An open general-purposes research system for automating the development and application of information technologies in the area of image processing, analysis, and evaluation. *Pattern Recognition and Image Analysis: Advances in Mathematical Theory and Applications*, 16(4), 530–563.

[33] Gurevich, I.B. and Koryabkina, I.V. (2006) Comparative analysis and classification of features for image models. *Pattern Recognition and Image Analysis: Advances in Mathematical Theory and Applications*, 16(3), 265–297.

[34] Gurevich, I. and Koryabkina, I. (2009) Method for image transform selection in cytological image analysis. In Gurevich, I., Niemann, H., and Salvetti, O. (eds.) *Image Mining Theory and Applications: Proceedings of the 2nd International Workshop on Image Mining Theory and Applications,* pp. 100–106.

[35] Gurevich, I.B. and Koryabkina, I.V. (2010) Selection of image transformations in the computer analysis of cytological specimens. *Pattern Recognition and Image Analysis: Advances in Mathematical Theory and Applications*, 20(1), 73–80.

[36] Gurevich, I.B. and Nefyodov, A.V. (2001) Algorithms for estimate calculations designed for 2D support sets. Part 1: Rectangular support sets. *Pattern Recognition and Image Analysis: Advances in Mathematical Theory and Applications*, 11(4), 662–689.

[37] Gurevich, I.B. and Nefyodov, A.V. (2005) Block diagram representation of a 2D-AEC algorithm with rectangular support sets. *Pattern Recognition and Image Analysis: Advances in Mathematical Theory and Applications*, 15(1), 187–191.

[38] Gurevich, I.B., Salvetti, O., and Trusova, Y.O. (2009) Fundamental concepts and elements of image analysis ontology. *Pattern Recognition and Image Analysis: Advances in Mathematical Theory and Applications*, 19(4), 603–611.

[39] Gurevich, I., Trusova, Y., and Yashina, V. (2013) Current trends in the algebraic image analysis. A survey. *Proceedings of 18th Iberoamerican Congress the Progress in Pattern Recognition, Image*

Analysis, Computer Vision, and Applications, Part I, LNCS, Vol. 8258, pp. 423–430.

[40] Gurevich, I.B. and Yashina, V.V. (2003) Descriptive image algebras with one ring. *Pattern Recognition and Image Analysis: Advances in Mathematical Theory and Applications*, 13(4), 579–599.

[41] Gurevich, I.B. and Yashina, V.V. (2005) Algorithmic scheme based on a descriptive image algebra with one ring: Image analysis example. *Pattern Recognition and Image Analysis: Advances in Mathematical Theory and Applications*, 15(1), 192–194.

[42] Gurevich, I. and Yashina, V. (2005) Generating descriptive trees. In Greiner, G., Hornegger, J., Niemann, H., and Stamminger, M. (eds.) *Proceeding "Vision, Modeling, and Visualization 2005"*, pp. 367–374.

[43] Gurevich, I.B. and Yashina, V.V. (2006) Operations of descriptive image algebras with one ring. *Pattern Recognition and Image Analysis: Advances in Mathematical Theory and Applications*, 16(3), 298–328.

[44] Gurevich, I.B. and Yashina, V.V. (2006) Computer-aided image analysis based on the concepts of invariance and equivalence. *Pattern Recognition and Image Analysis: Advances in Mathematical Theory and Applications*, 16(4), 564–589.

[45] Gurevich, I.B. and Yashina, V.V. (2008) Descriptive approach to image analysis: Image models. *Pattern Recognition and Image Analysis: Advances in Mathematical Theory and Applications*, 18(4), 518–541.

[46] Gurevich, I.B. and Yashina, V.V. (2012) Descriptive approach to image analysis: Image formalization space. *Pattern Recognition and Image Analysis: Advances in Mathematical Theory and Applications*, 22(4), 495–518.

[47] Gurevich, I.B. and Yashina, V.V. (2017) Descriptive image analysis. Genesis and current trends. *Pattern Recognition and Image Analysis: Advances in Mathematical Theory and Applications*, 27(4), 653–674.

[48] Gurevich I.B., Yashina V.V., Ablameyko S.V., Nedzved A.M., Ospanov A.M., Tleubaev A.T., Fedorov A.A., and Fedoruk N.A. (2018) Development and experimental investigation of mathematical methods for automating the diagnostics and analysis of ophthalmological images. *Pattern Recognition and Image Analysis*, 28(4), 612–636.

[49] Gurevich, I.B. and Yashina, V.V. (2019) Descriptive image analysis. Foundations and descriptive image algebras. *International Journal of Pattern Recognition and Artificial Intelligence*, 33(12), 1940018-1–1940018-25.

[50] Gurevich, I.B. and Yashina, V.V. (2019) Algebraic interpretation of image analysis operations. *Pattern Recognition and Image Analysis*, 29(3), 389–403.

[51] Gurevich, I.B. and Yashina, V.V. (2019) Descriptive image analysis: Part II. Descriptive image models. *Pattern Recognition and Image Analysis: Advances in Mathematical Theory and Applications*, 29(4), 598–612.

[52] Gurevich, I.B. and Yashina, V.V. (2019) Irrelation of mathematical and functional aspects of descriptive image algebras with one ring operations interpretability. In Ablameyko, S., Krasnoproshin, V., and Lukashevich, M. (eds.) *Pattern Recognition and Information Processing. PRIP 2019*. Communications in Computer and Information Science, Vol. 1055, pp. 74–85.

[53] Gurevich, I.B. and Yashina, V.V. (2020) Descriptive image analysis: III. Multilevel model for algorithms and initial data combining in pattern recognition. *Pattern Recognition and Image Analysis: Advances in Mathematical Theory and Applications*, 30(3), 328–341.

[54] Gurevich, I.B. and Yashina, V.V. (2020) Descriptive image analysis: Part IV. Information structure for generating descriptive algorithmic schemes for image recognition. *Pattern Recognition and Image Analysis: Advances in Mathematical Theory and Applications*, 30(4), 649–665.

[55] Gurevich, I. and Yashina, V. (2021) Basic models of descriptive image analysis. In Del Bimbo, A. *et al.* (eds.) *ICPR 2020 Workshops*, LNCS, Vol. 12665, pp. 275–288.

[56] Gurevich, I.B. and Yashina, V.V. (2021) Descriptive models of information transformation processes in image analysis. *Pattern Recognition and Image Analysis*, 31(3), 402–420.

[57] Gurevich, I.B., Zhuravlev, Y.I., Myagkov, A.A., Trusova, Y.O., and Yashina, V.V. (2015) On basic problems of image recognition in neurosciences and heuristic methods for their solution. *Pattern Recognition and Image Analysis: Advances in Mathematical Theory and Applications* (Pleiades Publishing), 25(1), 132–160.

[58] Hadwiger, H. (1950) Minkowskische addition and Subtraktion beliebiger Punktmengen und die Theoreme von Erhard Schmidt. *Mathematische Zeitschrift*, 53(3), 210–218.

[59] Haralick, R. and Shapiro L. (1985) Image segmentation techniques. *Computer Vision, Graphics, and Image Processing*, 29, 100–132.

[60] Haralick, R., Shapiro, L., and Lee, J. (1987) Morphological edge detection. *IEEE Journal on Robotics and Automation*, RA-3(1), 142–157.

[61] Haralick, R.M., Sternberg, S.R., and Zhuang, X. (1987) Image analysis using mathematical morphology. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, PAMI-9(4), 532–550.

[62] Joo, H., Haralick, R.N., and Shapiro, L.G. (1990) Toward the automatic generation of mathematical morphology procedures using predicate logic. *Proceedings of the Third International Conference on Computer Vision*, pp. 156–165.

[63] Kaneff, S. (1972) Pattern cognition and the organization of information. In Watanabe, S. (ed.) *The Proceedings of the International Conference on Frontiers of Pattern Recognition*, pp. 193–222.

[64] Kittler, J., Hatef, M., Duin, R., and Matas, J. (1998) On combining classifiers. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 20(3), 226–239.

[65] Kirsh, R. (1964) Computer interpretation of English text and picture patterns. *IEEE-TEC*, EC-13(4), 363–376.

[66] Köthe, U., Montanvert, A., and Soille, P. (eds.) (2012) *Applications of Discrete Geometry and Mathematical Morphology.* Lecture Notes in Computer Science, Vol. 7346, Springer, Berlin, Heidelberg.

[67] Kuzmin, E.N. (1975) On the Nagata-Higman theorem, mathematical structures, computational mathematics. *Mathematical Modeling. Proceedings Dedicated to the Sixtieth Birthday of Academician L. Iliev*, pp. 101–107.

[68] Labunetc, V.G. (1984) *Algebraic Theory of Signals and Systems (Digital Signal Processing)* (Krasnoyarsk University, Krasnoyarsk) [in Russian].

[69] Lam, L. and Suen, C.Y. (1995) Optimal combinations of pattern classifiers. *Pattern Recognition Letters*, 16, 945–954.

[70] Lam, L. and Suen, C.Y. (1997) Application of majority voting to pattern recognition: An analysis of its behavior and performance. *IEEE Transactions on Systems, Man, and Cybernetics, Part A*, 27, 553–568.

[71] Lee, J.S.J., Haralick, R.M., and Shapiro, L.G. (1987) Morphologic edge detection. *IEEE Journal of Robotics and Automation*, RA-3(2), 142–156.

[72] Leites, D.A. (1980) Introduction to the theory of supermanifolds. *Russian Mathematic Survey*, 35(1), 1–64.

[73] Maltsev, Y.N. and Kuzmin, E.N. (1978) A basis for identities of the algebra of second order matrices over a finite field. *Algebra Logic*, 17, 17–21.

[74] Malcev, A.I. (1973) *Algebraic Systems* (Springer-Verlag, Berlin).

[75] Maragos, P. (2003) Algebraic and PDE approaches for lattice scale-spaces with global constraints. *International Journal of Computer Vision*, 52(2/3), 121–137.

[76] Maragos, P. and Schafer, R. (1986) Morphological skeletons representation and coding of binary images. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, 34(5), 1228–1244.

[77] Maragos, P. and Schafer, R. (1987) Morphological filters Part I: Their set-theoretic analysis and relations to linear shift-invariant filters. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, 35, 1153–1169.

[78] Maragos, P. and Schafer, R. (1987) Morphological filters Part II: Their relations to median, order-statistic, and stack filters. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, 35, 1170–1184.

[79] Marr, D. (1982) *Vision* (W.H. Freeman, San Francisco).

[80] Matheron, G. (1975) *Random Sets and Integral Geometry* (Wiley, New York).

[81] Matrosov, V.L. (1983) Pair isomorphism of permissible objects in recognition problems. *Computational Mathematics and Mathematical Physics*, 23(1), 123–127.

[82] Matrosov, V.L. (1983) On the incompleteness of a model of algorithms for computing estimates. *Computational Mathematics and Mathematical Physics*, 23(2), 128–136.

[83] Matrosov, V.L. (1984) Lower bounds of the capacity of L-dimensional algebras of estimate-computing algorithms. *Computational Mathematics and Mathematical Physics*, 24(6), 182–188.

[84] Matrosov, V.L. (1985) The capacity of polynomial expansions of a set of algorithms for calculating estimates. *Computational Mathematics and Mathematical Physics*, 25(1), 79–87.

[85] Mazurov, V.D. (1971) Committees of inequalities systems and the recognition problem. *Cybernetics*, 7(3), 559–567.

[86] Mazurov, V.D. and Khachai, M.Y. (2004) Committees of systems of linear inequalities. *Automation and Remote Control*, 65(2), 193–203.

[87] Mazurov, V.D. and Khachai, M.Y. (2007) Parallel computations and committee constructions. *Automation and Remote Control*, 68(5), 912–921.

[88] Miagkov, A.A. and Yashina, V.V. (2009) Systematization and feature selection for formalization of descriptions of the methodological structure of cytological and histological preparations and analytical review. *Pattern Recognition and Image Analysis: Advances in Mathematical Theory and Applications*, 19(4), 673–678.

[89] Minkowski, H. (1901) Über die Konzepte von Länge, Fläche und Volumen. *Jahresbericht der Deutschen Mathematiker Vereinigung*, 9, 115–121.

[90] Narasimhan, R. (1966) Syntax-directed interpretation of classes of pictures. *Community ACM*, 9(3), 166–173.

[91] Narasimhan, R. (1970) Picture languages. In Kaneff, S. (ed.) *Picture Language Machines*, pp. 1–30.

[92] Neumann, J. (1951) The general logical theory of automata. *Celebral Mechanism in Behavior: The Hixon Symposium* (John Wiley & Sons, New York).

[93] Neumann, J. (1966) *Theory of Self-Reproducing Automata* (University of Illinois Press, Urbana).

[94] Pavel, M. (1976) Pattern recognition categories. *Pattern Recognition*, 8(3), 115–118.

[95] Pavel, M. (1989) *Fundamentals of Pattern Recognition* (Marcell, Dekker, Inc., New York).

[96] Pavlidis, T. (1977) Classification, Description and Syntactic Analysis. In: *Structural Pattern Recognition. Springer Series in Electrophysics*, Vol. 1. Springer, Berlin, Heidelberg.

[97] Pytiev, Y.P. (2004) *Method of Mathematical Modeling of Measuring and Computing Systems,* 2nd Ed. (MAIK Nauka, Moscow) [in Russian].

[98] Radunacu, B., Grana, M., and Albizuri, F.X. (2003) Morphological scale spaces and associative morphological memories: Results on robustness and practical applications. *Journal of Mathematical Imaging and Vision*, 19, 113–131.

[99] Rastrigin, L.A. and Erenshtein, R.K. (1975) Decision processes by a collection of decision rules in pattern recognition problems. *Automated Remote Control*, 36(9), 1503–1512.

[100] Rosenfeld, A. (1979) Digital topology. *American Mathematical Monthly*, 86.

[101] Rosenfeld, A. (1979) *Picture Languages. Formal Models for Picture Recognition* (Academic Press, New York).

[102] Ritter, G.X. and Wilson, J.N. (2001) *Handbook of Computer Vision Algorithms in Image Algebra, 2nd Ed.* (CRC Press Inc., Boca Raton).

[103] Ritter, G.X. (2001) *Image Algebra* (Center for Computer Vision and Visualization, Department of Computer and Information science and Engineering, University of Florida, Gainesville, FL 32611).

[104] Ritter, G.X. and Gader, P.D. (1987) Image algebra techniques for parallel image processing. *Parallel Distributed Computers*, 4(5), 7–44.

[105] Ritter, G.X., Wilson, J.N., and Davidson, J.L. (1990) Image algebra: An overview. *Computer Vision, Graphics, and Image Processing*, 49, 297–331.
[106] Rudakov, K.V. (1987) Universal and local constraints in the problem of correction of heuristic algorithms. *Cybernetics*, 23(2), 181–186.
[107] Rudakov, K.V. (1987) Completeness and universal constraints in the correction problem for heuristic classification algorithms. *Cybernetics*, 23(3), 414–418.
[108] Rudakov, K.V. (1987) Symmetric and functional constraints in the correction problem of heuristic classification algorithms. *Cybernetics*, 23(4), 528–533.
[109] Rudakov, K.V. (1988) Application of universal constraints in the analysis of classification algorithms. *Cybernetics*, 24(1), 1–6.
[110] Rudakov, K.V. and K. V. Vorontsov (1999) Methods of optimization and monotone correction in the algebraic approach to the recognition problem. *Doklady Mathematics*, 60, 139–142.
[111] Rudakov, K.V. and Chekhovich, Y.V. (2005) Completeness criteria for classification problems with set-theoretic constraints. *Computational Mathematics and Mathematical Physics*, 45(2), 329–337.
[112] Rudakov, K.V., Cherepnin, A.A., and Chekhovich, Y.V. (2007) On metric properties of spaces in classification problems. *Doklady Mathematics*, 76(2), 790–793.
[113] Rudakov, K.V. and Torshin, I.Y. (2011) Selection of informative feature values on the basis of solvability criteria in the problem of protein secondary structure recognition. *Doklady Mathematics*, 84(3), 871–874.
[114] Sauvola, J. and Pietikainen, M. (2000) Adaptive document image binarization. *Pattern Recognition*, 33(2), 225–236.
[115] Serra, J. (1982) *Image Analysis and Mathematical Morphology* (Academic Press, London).
[116] Serra, J. (1994) Morphological filtering: An overview. *Signal Processing*, 38(1), 3–11.
[117] Serra, J. (1998) Introduction to morphological filters. *Image Analysis and Mathematical Morphology*, 2, 101–114.
[118] Shapiro, L.G. and Stockman, G.C. (2001). *Computer Vision* (Prentice Hall, Englewood).
[119] Schlesinger, M. and Hlavac, V. (2002) Ten lectures on statistical and structural pattern recognition. *Computational Imaging and Vision*, Vol. 24 (Kluwer Academic Publishers, Dordrecht).
[120] Shaw, A. (1967) A Proposed Language for the Formal Description of Pictures. CGS Memo. 28, Stanford University.

[121] Sinicyn, N. (2007) *Calman and Pugachev Filters,* 2nd Ed. (Logos, Moscow) [in Russian].

[122] Soille, P. (1996) Morphological partitioning of multispectral images. *Journal of Electronic Imaging*, 5(3), 252–265.

[123] Soille, P. (2003, 2004) *Morphological Image Analysis. Principles and Applications, 2nd Ed.* (Springer-Verlag, Berlin).

[124] Sternberg, S.R. (1980) Language and architecture for parallel image processing. *Proceedings of the Conference on Pattern Recognition in Practice.*

[125] Sternberg, S.R. (1985) An overview of image algebra and related architectures. In Levialdi, S. (ed.) *Integrated Technology for Parallel Image Processing* (Academic Press, London).

[126] Sternberg, S.R. (1986) Grayscale morphology. *Computer Vision, Graphics and Image Processing*, 35(3), 333–355.

[127] Suen, C.Y., Nadal, C.P., Mai, T.A., Legault, R., and Lam, L. (1990) Recognition of totally unconstrained handwritten numerals based on the concept of multiple experts. In Suen, C.Y. (ed.) *Frontiers in Handwriting Recognition*, pp. 131–143.

[128] Suen, C.Y., Nadal, C.P., Legault, R., Mai, T.A., and Lam, L. (1992) Computer recognition of unconstrained handwritten numerals. *Proceedings of the IEEE*, 80(7), 1162–1180.

[129] Suen, C.Y. and Lam, L. (2000) Multiple classifier combination methodologies for different output levels. *Multiple Classifier Systems.* Lecture Notes in Computer Science, Vol. 1857, pp. 52–66.

[130] Sussner, P. (2000) Observations on morphological associative memories and the kernel method. *Neurocomputing*, 31, 167–183.

[131] Sussner, P. and Ritter, G.X. (2000) Rank-based decompositions of morphological templates. *IEEE Transactions on Image Processing*, 9(8), 1420–1430.

[132] Sussner, P. (2003) Generalizing operations of binary autoassociative morphological memories using fuzzy set theory. *Journal of Mathematical Imaging and Vision*, 19, 81–93.

[133] Tax, D.M.J. and Duin, R.P.W. (2001) Combining one-class classifiers. *Proceedings of the Second International Workshop Multiple Classifier Systems.*

[134] Toet, A. (1989) A morphological pyramidal image decomposition. *Pattern Recognition Letters*, 9, 255–261.

[135] Unger, S.H. (1958) A computer oriented toward spatial problems. *Proceedings of the IRE*, 46, 1744–1750.

[136] Van Der Waerden, B.L. (1971) *Algebra I, Algebra II* (Springer-Verlag, Berlin).

[137] Winbridge, D. and Kittler, J. (2001) Classifier combination as a tomographic process. *Proceedings of the Second International Workshop Multiple Classifier Systems*, pp. 248–258.

[138] Zhuravlev, Y.I. (1978) On algebraic approach for solving the recognition and classification problems. *Problems of Cybernetics*, 33 [in Russian].

[139] Zhuravlev, Y.I. (1998) An algebraic approach to recognition and classification problems. *Pattern Recognition and Image Analysis: Advances in Mathematical Theory and Applications*, 8, 59–100.

© 2025 World Scientific Publishing Company
https://doi.org/10.1142/9789811267215_0002

Chapter 2

Morphological Image Analysis: Mathematical Foundations and Applications

Yuri P. Pyt'ev*, Alexey I. Chulichkov†, Olesya V. Falomkina‡, Dmitriy A. Balakin§, and Andrey V. Zubyuk¶

M. V. Lomonosov Moscow State University, Moscow, the Russian Federation

**yuri.pytyev@gmail.com*

†*achulichkov@gmail.com*

‡*olesya.falomkina@gmail.com*

§*balakin_d_a@physics.msu.ru*

¶*zubjuk@physics.msu.ru*

A short review of mathematical methods of image analysis is provided. Mathematical models of shapes of signals and images are formulated, and mathematical methods and numerical algorithms for calculating characteristics of closeness (connectedness and dependence) of two signal or image shapes using oblique projection technique are considered. Examples of using morphological image analysis to solve applied problems are given. In addition, elements of mathematical formalism of subjective modeling of categories of fuzziness and uncertainty that reflect incompleteness and unreliability of information used in construction of a

subjective morphological model to take into account researcher's subjective notions. As their applications, the mathematical foundations of morphological image analysis, oblique projection approach and analysis of dependency of relative image shapes and connectedness of their absolute shapes are shown to have an interpretation in terms of subjective modeling, with examples of subjective morphological models and optimization of decisions based on them using oblique projection and subjective optimization. It is shown that optimal solutions of morphological problems of image analysis and image interpretation can be formulated in terms of oblique image projection and subjective optimization.

Keywords: Morphological analysis, image analysis, image recognition, image identification, discovery of emergent unknown objects, two-scale image model, image shape, image equivalent, shape-determining, projection operator, shape in a broad sense, mosaic image, morphological proximity, rival similarity, morphological dependency, connectedness, relative shape, oblique projection, comparative analysis, absolute shape, subjective morphological model, morphological dependency and connectedness, plausibility and belief measures, modalities of researcher's subjective judgments, problem of optimum estimation, error plausibility, mutually morphologically independent shapes, connectedness of shapes, absolute shape, non-local method for object extraction.

2.1 Introduction

Morphological methods of image analysis make up a vast field in which different approaches are used. Common in these approaches is that image analysis is based on mathematical models of brightness-geometric features of the image.

There are several ways to describe the brightness-geometric features of images or their fragments.

The simplest one is the decomposition of an image or its fragment using a system of basis functions (Fourier, Haar, and Walsh) [15], wavelet transform [17,23], Hough transform [14], etc. Image features appear as maxima of decomposition coefficients for the chosen system of functions. The extraction of brightness-geometric features reduces to linear filtering. As a result of this filtering, only bright-geometric features that are linear combinations of basis functions remain in the image.

Nonlinear filtering, for example, median filtering and its modifications, is also used to extract brightness-geometric features of images that are distorted by impulse noise [12].

One of the most widely known approaches to image brightness-geometric structure analysis is mathematical morphology proposed by Serra [38]. It was first applied to binary images. Its main objective was to identify the components of a binary image that satisfy specified geometric shape and size constraints. Mathematical morphology offers a set of image transformations aimed at extracting significant image features represented by a structuring element which is a binary image as well but usually is considerably smaller. The most common operations are dilation, erosion, opening, and closing. Later, methods of binary morphology were generalized for gray-scale images [3, 7, 16, 40].

As an extension of mathematical morphology, Maragos proposed a spectral approach to describe the shapes of plane figures [20]. The morphological spectrum of an image is defined as the derivative of the area of its opening w.r.t. the scale parameter of a structuring element. This approach was developed in publications [4, 41–43].

Another approach to extracting brightness-geometric features is based on integral transformations with diffusion-type kernels or heat kernels. The integral operator is chosen so that the transformation result preserves specified image fragment features and suppresses noise [19]. The terms "heat kernel" and "heat diffusion" were introduced in [2]. Diffusion operator theory was developed in [4, 5, 18]. Descriptions of image shapes related to spectral characteristics of diffusion operator were suggested in [1, 6, 13, 39, 41].

If an image of a distinct object on a background is analyzed, then the geometric features of this object can be described by its contour. The parameters of this contour depend on the object's shape and can be useful for analyzing the object by its image. The contour of the object image can be specified by its skeleton. The image skeleton is the set of points equidistant from the contour. The skeleton is considered as a graph. Its vertex locations, edge length, and curvature allow one to solve a number of problems of computer vision, recognition, and classification of geometric shapes of distinct objects presented in the image [21, 22].

Almost all approaches listed above are characterized by the lack of a rigorous image shape definition. In contrast, morphological analysis methods presented in this chapter are based on a rigorous definition of the image shape and a fairly general mathematical model of image registration.

An overview and comparative analysis of morphological methods for solving image analysis and interpretation problems is given in [9, 11, 26–31, 36, 37, 45] and in the monograph [35].

A general image registration model assumes that the researcher knows the set of registration parameters, such as lighting conditions, its spectral composition, registration equipment characteristics, optical characteristics of the scene, and others. Specific values of registration parameters for a given image are unknown. However, the model allows us to determine how the scene image changes if the conditions of its registration change. Thus, each scene is associated with the set of its images obtained under all possible registration conditions; this set of images is called the shape of the scene image.

This approach allows solving a number of problems of image analysis and interpretation. For example, to recognize and identify scenes from their images, it is required to determine whether the presented image belongs to the shape of scene images or to find the shape closest to the presented image. The same image operations have to be carried out to solve a different problem, in which two scene images are registered under different and unknown conditions, and the objective is to determine whether they are images of the same scene. If this is not the case, it is required to extract the differences between images which cannot occur due to the change in registration conditions and, therefore, are caused by the change of the scene itself. In addition, the proposed approach allows the use of subjective perceptions of the researcher [25, 32–34] for formulation of image registration model and for obtaining the optimal solution to morphological problems. The used formalism of subjective modeling significantly differs from the widely known approaches [8, 44] and their applications to image morphology [10, 24].

Of course, some specific image shape models lead to approaches similar to known ones, but the approach itself seems original and very fruitful.

2.2 Elements of Morphological Analysis [33, 35]

As a mathematical object, an image is a vector-valued function $\boldsymbol{f}(\cdot) : X \to R^N$, where R^N is an N-dimensional Euclidean space, while X, called the *field of view,* is a bounded closed domain on plane R^2 or its discrete representation by a subset $\{1, 2, \ldots\}^2$. The norm $\|\boldsymbol{f}(x)\|_N$ is the *brightness of image* $\boldsymbol{f}(\cdot)$ *at point* $x \in X$, where $\|\boldsymbol{z}(x)\|_N^2 = z_1^2 + \cdots + z_N^2$ and $\boldsymbol{z} = (z_1, \ldots, z_N) \in R^N$, and $\boldsymbol{f}(x)\|\boldsymbol{f}(x)\|_N$ is called the color of $\boldsymbol{f}(\cdot)$ at this point. The norm is induced by the scalar product $(\boldsymbol{z}^1, \boldsymbol{z}^2)_N = z_1^1 z_1^2 + \cdots + z_N^1 z_N^2$. Linear operations $(\boldsymbol{f}_1 + \boldsymbol{f}_2)(\cdot)$ and $(a \cdot \boldsymbol{f})(\cdot), a \in R^1$, on images are defined as follows: $(\boldsymbol{f}_1 + \boldsymbol{f}_2)(x) = \boldsymbol{f}_1(x) + \boldsymbol{f}_2(x)$ and $(a \cdot \boldsymbol{f})(x) = a \cdot \boldsymbol{f}(x), x \in X$.

Finally, in the Borel class of R^2, a measure μ is defined so that X and any image $\boldsymbol{f}(\cdot)$ are μ-measurable, and any function $\|\boldsymbol{f}(x)\|_N^2$ is μ-integrable. The set of images, denoted by $\vec{L}_\mu^2(X)$, is a Euclidean space, in which

$$\|\boldsymbol{f}\|^2 = \int_X \|\boldsymbol{f}(x)\|_N^2 \mu(dx) < \infty,$$

$$(\boldsymbol{f}, \boldsymbol{g}) = \int_X (\boldsymbol{f}(x), \boldsymbol{g}(x))_N \mu(dx) < \infty$$

are the norm of the image $\boldsymbol{f}(\cdot)$ and the scalar product of the images $\boldsymbol{f}(\cdot)$ and $\boldsymbol{g}(\cdot)$, respectively; moreover, $\|\boldsymbol{f}(\cdot)\| \in L_\mu^2(X)$.

2.2.1 *Image shape and shape of a scene image class*

Let G denote a scene, K be a set of scene image registration conditions, such as weather conditions, lighting conditions, and other conditions that are unimportant to the researcher, such as the reflectivity of the elements of G, and let $\vec{V}_G = \{\boldsymbol{f}(\cdot, k) \in \vec{L}_\mu^2(X), k \in K\}$ denote the class of images of G that can be obtained under any possible registration conditions $k \in K$.

The class $\vec{V}_G$ carries complete information about geometric characteristics of G and about the shapes of its elements, which are represented by images $\boldsymbol{f}(\cdot, k), k \in K$.

Given an image $\boldsymbol{f}(\cdot) \in \vec{V}_G$ we define the class

$$\vec{V}_G(\boldsymbol{f}) = \{\boldsymbol{g}(\cdot) \in \vec{V}_G, \boldsymbol{g}(\cdot) \preccurlyeq \boldsymbol{f}(\cdot)\} \tag{2.1}$$

of images $\boldsymbol{g}(\cdot) \in \vec{V}_G$ that represent geometric characteristics of G in no more detail than $\boldsymbol{f}(\cdot)$.

The class $\vec{V}_G(\boldsymbol{f})$ completely describes the relative quality of image $\boldsymbol{f}(\cdot)$ as a visual representation of the geometric characteristics of G compared to images $\boldsymbol{g}(\cdot) \preccurlyeq \boldsymbol{f}(\cdot), \boldsymbol{g}(\cdot) \in \vec{V}_G$. We say that $\vec{V}_G(\boldsymbol{f})$ is the *G-shape of image* $\boldsymbol{f}(\cdot)$ or merely the shape of $\boldsymbol{f}(\cdot)$ if it is clear from context the shape of which scene (referred to hereafter as *scene shape*) is represented by $\boldsymbol{f}(\cdot)$. Accordingly, the class $\vec{V}_G$ is called the *shape of images* of G.

In equation (2.1), the relation $\preccurlyeq$ means that an image $\boldsymbol{g}(\cdot)$ is comparable (in shape; hereafter, the words "in shape" are omitted) with $\boldsymbol{f}(\cdot)$ and it is not more detailed than $\boldsymbol{f}(\cdot)$. Hence, it may not represent all details of G that are represented by $\boldsymbol{f}(\cdot)$. For example, if $\boldsymbol{g}(x) = F(\boldsymbol{f}(x)) \stackrel{\text{def}}{=} F \circ \boldsymbol{f}(x), x \in X$, where $F(\cdot) : R^N \to R^N$ and $\|F \circ \boldsymbol{f}\| < \infty$, then it is possible that $\boldsymbol{g}(x_1) = \boldsymbol{g}(x_2)$, even though $\boldsymbol{f}(x_1) \neq \boldsymbol{f}(x_2)$. We say that image $\boldsymbol{f}(\cdot)$ and any image $\boldsymbol{g}(\cdot) \in \vec{V}_G(\boldsymbol{f})$ are comparable. Moreover, any image $\boldsymbol{g}(\cdot) \in \vec{V}_G(\boldsymbol{f})$ is not more detailed than $\boldsymbol{f}(\cdot)$ (written as $\boldsymbol{g}(\cdot) \preccurlyeq \boldsymbol{f}(\cdot)$).

The relation $\preccurlyeq$ defines a partial quasiorder on the class $\vec{V}_G$. The relation $\preccurlyeq$ is reflexive ($\boldsymbol{g}(\cdot) \preccurlyeq \boldsymbol{g}(\cdot)$, $\boldsymbol{g}(\cdot) \in \vec{V}_G$) and transitive: relations $\boldsymbol{h}(\cdot) \preccurlyeq \boldsymbol{g}(\cdot)$ and $\boldsymbol{g}(\cdot) \preccurlyeq \boldsymbol{f}(\cdot)$ imply that $\boldsymbol{h}(\cdot) \preccurlyeq \boldsymbol{f}(\cdot)$. However, first, if $\boldsymbol{g}(\cdot) \preccurlyeq \boldsymbol{f}(\cdot)$ and $\boldsymbol{f}(\cdot) \preccurlyeq \boldsymbol{g}(\cdot)$, then, generally speaking, $\boldsymbol{g}(\cdot) \neq \boldsymbol{f}(\cdot)$, and, second, an arbitrary pair of images $\boldsymbol{f}(\cdot), \boldsymbol{g}(\cdot)$ from $\vec{V}_G$ may not be comparable, i.e. neither $\boldsymbol{g}(\cdot) \preccurlyeq \boldsymbol{f}(\cdot)$ nor $\boldsymbol{f}(\cdot) \preccurlyeq \boldsymbol{g}(\cdot)$ is true (these conditions characterize the relative degree of detail about G represented in $\boldsymbol{f}(\cdot)$ and $\boldsymbol{g}(\cdot)$). The same is true for an arbitrary pair of images $\boldsymbol{g}_1(\cdot)$, $g_2(\cdot)$ from $\vec{V}_G(\boldsymbol{f})$.

However, if $\boldsymbol{g}(\cdot) \preccurlyeq \boldsymbol{f}(\cdot)$ and $\boldsymbol{f}(\cdot) \preccurlyeq \boldsymbol{g}(\cdot)$, then images $\boldsymbol{f}(\cdot)$ and $\boldsymbol{g}(\cdot)$ are equivalent, i.e. they represent the shape of G to an identical degree of detail (written as $\boldsymbol{f}(\cdot) \Leftrightarrow \boldsymbol{g}(\cdot)$). Moreover, if $\boldsymbol{f}(\cdot) \Leftrightarrow \boldsymbol{g}(\cdot)$, then $\boldsymbol{g}(\cdot) \Leftrightarrow \boldsymbol{f}(\cdot)$, and, additionally, if $\boldsymbol{g}(\cdot) \Leftrightarrow \boldsymbol{h}(\cdot)$, then $\boldsymbol{f}(\cdot) \Leftrightarrow \boldsymbol{h}(\cdot)$. Consequently, the relation $\Leftrightarrow$ is an equivalence relation.

Therefore, each image $\boldsymbol{f}(\cdot) \in \vec{V}_G$ is associated with the class $\vec{E}_G(\boldsymbol{f})$ of images that are equivalent to $\boldsymbol{f}$, are mutually equivalent,

and represent G to an identical degree of detail:

$$\vec{E}_G(\boldsymbol{f}) = \{\boldsymbol{g}(\cdot) \in \vec{V}_G, \boldsymbol{g}(\cdot) \Leftrightarrow \boldsymbol{f}(\cdot), \boldsymbol{f}(\cdot) \in \vec{V}_G\}.$$

The partial quasiordering $\preccurlyeq$ defined on $\vec{V}_G$ generates a partial ordering $\leq$ on the set $\vec{E}_G$ of classes $\vec{E}_G(\boldsymbol{f}), \boldsymbol{f}(\cdot) \in \vec{V}_G$, of equivalent images: for any $\boldsymbol{f}_1 \in \vec{E}_1$ and $\boldsymbol{f}_2 \in \vec{E}_2, \vec{E}_1 \leq \vec{E}_2$ if $\boldsymbol{f}_1(\cdot) \preccurlyeq \boldsymbol{f}_2(\cdot)$. On $\vec{E}_G$, the relation $\leq$ is transitive and reflexive, and the conjunction $\vec{E}_1 \leq \vec{E}_2$ and $\vec{E}_2 \leq \vec{E}_1$ is equivalent to the identity of $\vec{E}_1$ and $\vec{E}_2 : \vec{E}_1 \equiv \vec{E}_2$.

As a rule, for each scene G, we can indicate a minimal (by inclusion) class $\vec{J}_G \subset \vec{V}_G$ of its most detailed mutually incomparable images $\boldsymbol{f}(\cdot) \in \vec{V}_G$ such that $\vec{E}_G(\boldsymbol{f}^{(1)}) \cap \vec{E}_G(\boldsymbol{f}^{(2)}) = \varnothing$ if

$$\mu(\{x \in X, \boldsymbol{f}^{(1)}(x) \neq \boldsymbol{f}^{(2)}(x)\}) > 0, \quad \boldsymbol{f}^{(i)} \in \boldsymbol{J}_G, \quad i = 1, 2,$$

and $\vec{V}_G = \bigcup_{\boldsymbol{f}(\cdot) \in \boldsymbol{J}_G} \vec{V}_G(\boldsymbol{f})$; moreover, $\forall \boldsymbol{g} \in \vec{V}_G \, \exists \boldsymbol{f}(\cdot) \in \vec{J}_G$: $\boldsymbol{g}(\cdot) \preccurlyeq \boldsymbol{f}(\cdot)$.

The class $\vec{J}_G$ is said to determine the shape $\vec{V}_G$ of images of scene G and thus is called G-shape-determining. Therefore, the class $\vec{V}_G$ (which is the shape of images of G) is also the shape of image class $\vec{J}_G \subset \vec{V}_G$.

Any two images $\boldsymbol{g}_1(\cdot), \boldsymbol{g}_2(\cdot)$ from $\vec{V}_G(\boldsymbol{f})$ are said to be equivalent if there is $\alpha > 0$ such that $\boldsymbol{g}_1(\cdot) = \alpha \boldsymbol{g}_2(\cdot)$. In this case, for any $\alpha > 0$, $\vec{E}_G(\boldsymbol{f}) = \vec{E}_G(\alpha \boldsymbol{f})$ and $\vec{V}_G(\boldsymbol{f}) = \vec{V}_G(\alpha \boldsymbol{f})$; i.e. $\vec{E}_G(\boldsymbol{f})$ and $\vec{V}_G(\boldsymbol{f})$ are cones in $\vec{L}^2_\mu(X)$. Moreover, usually, if $\boldsymbol{g}_1(\cdot) \preccurlyeq \boldsymbol{f}(\cdot)$ and $\boldsymbol{g}_2(\cdot) \preccurlyeq \boldsymbol{f}(\cdot)$, then, for any $\alpha_1 > 0$ and $\alpha_2 > 0$, we have not only relations $\alpha_1 \boldsymbol{g}_1(\cdot) \leqslant \boldsymbol{f}(\cdot)$ and $\alpha_2 \boldsymbol{g}_2(\cdot) \leqslant \boldsymbol{f}(\cdot)$ but also $(\alpha_1 \boldsymbol{g}_1 + \alpha_2 \boldsymbol{g}_2)(\cdot) \leqslant \boldsymbol{f}(\cdot)$. The last condition means that for any $\boldsymbol{f}(\cdot) \in \vec{V}_G$ set $\vec{V}_G(\boldsymbol{f})$ is a convex cone in $\vec{L}^2_\mu(X)$. Finally, for any $\boldsymbol{f}(\cdot) \in \vec{V}_G$, the convex cone $\vec{V}_G(\boldsymbol{f})$ is usually a closed set in $\vec{L}^2_\mu(X)$, i.e. for any sequence $\boldsymbol{f}_i(\cdot) \in \vec{V}_G(\boldsymbol{f}), i = 1, 2, \ldots$, that converges in $\vec{L}^2_\mu(X)$, its limit $\lim_{i \to \infty} \boldsymbol{f}_i(\cdot) \in \vec{V}_G(\boldsymbol{f})$ (see [35]).

Under these conditions, there exists an operator $\mathbf{P}_{\boldsymbol{f}}$: $\vec{L}^2_\mu(X) \to \vec{L}^2_\mu(X)$ defined by the condition

$$\|\boldsymbol{g}(\cdot) - \mathbf{P}_{\boldsymbol{f}} \boldsymbol{g}(\cdot)\| = \inf_{\boldsymbol{h}(\cdot) \in \vec{V}_G(\boldsymbol{f})} \|\boldsymbol{g}(\cdot) - \boldsymbol{h}(\cdot)\|, \quad \boldsymbol{g}(\cdot) \in \vec{L}^2_\mu(X), \quad (2.2)$$

which is called the projection operator onto $\vec{V}_G(\boldsymbol{f})$. (Under these conditions, the infimum on the right-hand side of equation (2.2) is

attained at a unique image $\mathbf{P}_{\boldsymbol{f}}\boldsymbol{g} \in \vec{V}_G(\boldsymbol{f})$.) For any image $\boldsymbol{g}(\cdot) \in \vec{L}^2_\mu(X)$, the image $\mathbf{P}_{\boldsymbol{f}}\boldsymbol{g}(\cdot)$ is the nearest to $\boldsymbol{g}(\cdot)$ among all images in $\vec{L}^2_\mu(X)$ that belong to $\vec{V}_G(\boldsymbol{f})$, i.e. among all images that are less detailed than $\boldsymbol{f}(\cdot) \in \vec{V}_G$.

Since, under the above conditions, the operator $\mathbf{P}_{\boldsymbol{f}}$ defines $\vec{V}_G(\boldsymbol{f})$, i.e. $\vec{V}_G(\boldsymbol{f}) = \{\boldsymbol{g}(\cdot) \in \vec{L}^2_\mu(X), \mathbf{P}_{\boldsymbol{f}}\boldsymbol{g}(\cdot) = \boldsymbol{g}(\cdot)\}$, the operator $\mathbf{P}_{\boldsymbol{f}}$ is called the shape of image $\boldsymbol{f}(\cdot)$, or, more precisely, the (operator) representation of shape $\vec{V}_G(\boldsymbol{f})$ of image $\boldsymbol{f}(\cdot) \in \vec{V}_G$.

Let $\vec{L}_{\boldsymbol{f}}(X)$ be the inclusion-wise minimal subspace of $\vec{L}^2_\mu(X)$ that contains the cone $\vec{V}_G(\boldsymbol{f})$. If $\mathbf{\Pi}_{\boldsymbol{f}} : \vec{L}^2_\mu(X) \to \vec{L}^2_\mu(X)$ is the orthogonal projector onto $\vec{L}_{\boldsymbol{f}}(X), \hat{\mathbf{P}}_{\boldsymbol{f}} : \vec{L}_{\boldsymbol{f}}(X) \to \vec{L}_{\boldsymbol{f}}(X)$ is the orthogonal projector onto $\vec{V}_G(\boldsymbol{f})$, and $\mathbf{P}_{\boldsymbol{f}} : \vec{L}^2_\mu(X) \to \vec{L}^2_\mu(X)$ is the orthogonal projector onto shape $\vec{V}_G(\boldsymbol{f})$, then $\mathbf{P}_{\boldsymbol{f}}\boldsymbol{g}(\cdot) = \hat{\mathbf{P}}_{\boldsymbol{f}}\mathbf{\Pi}_{\boldsymbol{f}}\boldsymbol{g}(\cdot)$, although the sum of distances from image $\boldsymbol{g}(\cdot)$ to $\mathbf{\Pi}_{\boldsymbol{f}}\boldsymbol{g}(\cdot)$ and from $\mathbf{\Pi}_{\boldsymbol{f}}\boldsymbol{g}(\cdot)$ to $\mathbf{P}_{\boldsymbol{f}}\boldsymbol{g}(\cdot)$ is larger than the distance from image $\boldsymbol{g}(\cdot)$ to $\mathbf{P}_{\boldsymbol{f}}\boldsymbol{g}(\cdot)$ (projection transitivity lemma).

The subspace $\vec{L}_{\boldsymbol{f}}(X)$ and the orthogonal projector $\mathbf{\Pi}_{\boldsymbol{f}}$ representing it are called the shape of $\boldsymbol{f}(\cdot)$ in a broad sense. In case of a shape in a broad sense, the relations $\Leftrightarrow$ and $\preccurlyeq$ satisfy the following conditions: for any $\alpha \neq 0$, it is true that $\alpha\boldsymbol{f}(\cdot) \Leftrightarrow \boldsymbol{f}(\cdot)$, and, for any α_1 and α_2, it is true that $\alpha_1\boldsymbol{g}_1(\cdot) + \alpha_2\boldsymbol{g}_2(\cdot) \preccurlyeq \boldsymbol{f}(\cdot)$ if $\boldsymbol{g}_1(\cdot) \preccurlyeq \boldsymbol{f}(\cdot)$ and $\boldsymbol{g}_2(\cdot) \preccurlyeq \boldsymbol{f}(\cdot)$.

2.2.2 *Example of a gray-scale mosaic scene image shape*

Consider a gray-scale image shape example. For simplicity, $N = 1$, so $f(\cdot)$ is a scalar. To emphasize this, $f(\cdot)$ is used to denote images instead of $\boldsymbol{f}(\cdot)$. The Euclidean space of all gray-scale images is denoted by $L^2_\mu(X)$.

Let the class J_G of shape-determining image of a scene G consist of a single element $f(\cdot)$ given by

$$f(x) = \sum_{i=1}^{n} c_i \chi_i(x), \quad x \in X, \tag{2.3}$$

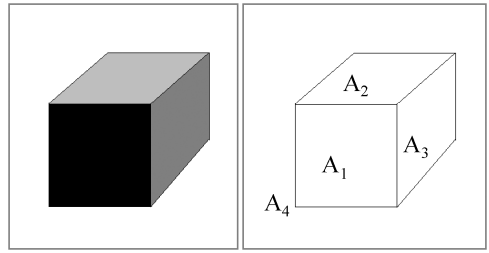


Figure 2.1. A mosaic image and the corresponding partition of the field of view into four disjoint subsets.

where $\chi_i(x) = 1$, if $x \in A_i, \chi_i(x) = 0$, if $x \notin A_i$, are characteristic functions of sets $A_1, \ldots, A_n$ that form a partition of the field of view X, i.e. $A_i \cap A_j = \varnothing$, for $i \neq j, i, j = 1, \ldots, n$, and $\bigcup_{i=1}^{n} A_i = X$. Such images are called mosaic.

A mosaic image example is given in Figure 2.1. Four areas of the field of view correspond to three visible cube faces and the background. Brightnesses of all three faces and the background differ.

Let us further assume that any change in registration conditions only affects brightness, so any scene image has the form $g(x) = F(f(x)) \stackrel{\text{def}}{=} F \circ f(x)$, where $F(\cdot) : R^1 \to R^1$ is a numerical function that models the brightness change of areas $A_1, \ldots, A_n$ of the field of view in the image of scene G under changed registration conditions. $F \in \boldsymbol{F}$, where $\boldsymbol{F}$ is the class of brightness transformations of image $f(\cdot)$ that model changes of scene registration. Then the set of all images of the scene G is the set

$$V_G(f) = \{g(\cdot) = F \circ f(\cdot),\, F \in \boldsymbol{F}\}. \tag{2.4}$$

In particular, if $\boldsymbol{F}$ is the class of all Borel functions, then

$$V_G(f) = \left\{ g(\cdot) = \sum_{i=1}^{n} b_j \chi_j(\cdot),\, b_i \in (-\infty, \infty), i = 1, \ldots, n \right\}. \tag{2.5}$$

In this example, the depicted scene is characterized by the geometrical shapes of areas $A_1, \ldots, A_n \subset X$, while constants $c_1, \ldots, c_n$ define the brightnesses of these areas.

Figure 2.2 shows some cube images of the form $g = F \circ f$. Images $g_0(\cdot)$ and $f(\cdot)$ are shape-equivalent, and $g_1(\cdot)$ and $g_2(\cdot)$ are less detailed than $f(\cdot)$ but are not equivalent to $f(\cdot)$. The uniformly bright

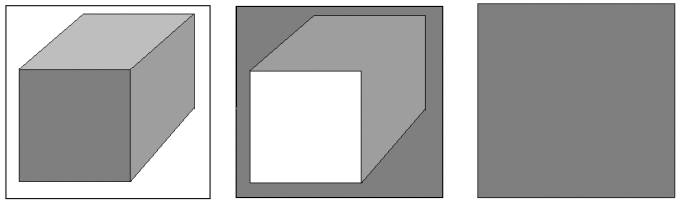


Figure 2.2. Images that are less detailed than the image in Figure 2.1; $g_i(\cdot) \prec = f(\cdot), i = 0, 1, 2; g_0(\cdot) \Leftrightarrow f(\cdot)$.

image $g_2(\cdot)$ is the least detailed one and it is comparable to a mosaic image of form (2.3) for any characteristic functions $\chi_1(\cdot), \ldots, \chi_n(\cdot)$, if the class $\boldsymbol{F}$ contains the function $F(z) = \text{const}\ \forall z \in R^1$.

The shape of images of the scene G (2.5) is an n-dimensional linear subspace of $L^2_\mu(X)$, and it biuniquely corresponds to the orthogonal projector $\mathbf{P}_f$ onto $V_G(f)$ that is defined for any $g \in L^2_\mu(X)$ as a solution of the best approximation problem

$$\|g - \mathbf{P}_f g\|^2 = \inf \{\|g - h\|^2 | \, h \in V_G(f)\}.$$

For the image shape of scene G given by equation (2.5),

$$\mathbf{P}_f g = \sum_{i=1}^{n} \frac{(g, \chi_i)}{\|\chi_i\|^2} \chi_i.$$

2.2.3 *Example of a gray-scale scene image shape as a convex closed cone*

Consider an example of a gray-scale image of scene G in which the class of functions $\boldsymbol{F}$ in equation (2.4) is the class of monotonously non-decreasing functions, and, as in the previous example, $J_G = \{f(\cdot)\}$. Then the shape $V_G(f)$ consists of images such that their local brightness maxima and minima coincide with the local extrema of the image $f(\cdot)$. The shape $V_G(f)$ is in this case a convex closed cone in the space $L^2_\mu(X)$.

This scene image shape model considers that registration conditions cannot affect the brightness ordering of areas $A_1, \ldots, A_n$. If the brightnesses of areas are ordered according to $c_1 < c_2 < \cdots < c_n$,

the shape of G is the convex closed cone

$$V_G(f) = \left\{ g(\cdot) = \sum_{i=1}^{n} b_j \chi_j(\cdot), b_1 \leq b_2 \leq \cdots \leq b_n \right\} \tag{2.6}$$

or the projector onto $V_G(f)$.

2.2.4 *Examples of morphological problem solutions*

2.2.4.1 *Scene recognition*

Let $\vec{V}_G(\boldsymbol{f})$ be the shape of an image of scene G represented by the projector $\mathbf{P}_{\boldsymbol{f}}$. For an image $\boldsymbol{g}(\cdot)$ containment $\boldsymbol{g}(\cdot) \in \vec{V}_G(\boldsymbol{f})$, equivalent to $\mathbf{P}_{\boldsymbol{f}}\boldsymbol{g} = \boldsymbol{g}$, means that under some registration conditions $\boldsymbol{g}(\cdot)$ can be an image of the scene G.

In practice, image registration is accompanied by some error. If $\boldsymbol{g}(\cdot)$ is the ideal image (undistorted by noise), the registration result is

$$\boldsymbol{\xi}(\cdot) = \boldsymbol{g}(\cdot) + \boldsymbol{\nu}(\cdot), \tag{2.7}$$

where $\boldsymbol{\nu}(\cdot)$ is the "noise image". In this case to decide whether $\boldsymbol{\xi}(\cdot)$ is a distorted image of the scene G, one has to test the hypothesis $\boldsymbol{g}(\cdot) \in \vec{V}_G(\boldsymbol{f})$ using the observation (2.7). For this test, it is convenient to use the test statistic

$$\tau_G(\boldsymbol{\xi}) = \frac{\|\boldsymbol{\xi} - \mathbf{P}_{\boldsymbol{f}}\boldsymbol{\xi}\|^2}{\|\mathbf{P_0}\boldsymbol{\xi} - \mathbf{P}_{\boldsymbol{f}}\boldsymbol{\xi}\|^2},$$

where $\mathbf{P}_0$ is the orthogonal projector onto the set of uniformly bright images, $\mathbf{P}_0\boldsymbol{\xi}$ is the image whose brightness at any point is equal to the average brightness of $\boldsymbol{\xi}(\cdot)$. The value of $\tau_G(\boldsymbol{\xi})$ is low for $\boldsymbol{\xi}(\cdot)$ close (in norm of $\vec{L}^2_\mu(X)$) to $\vec{V}_G(\boldsymbol{f})$, and it is high if the image is close to a uniformly bright image (close in shape to any scene image). Therefore, one tests $\tau_G(\boldsymbol{\xi}) \leq c_0$. If it is fulfilled, the hypothesis $\boldsymbol{g}(\cdot) \in \vec{V}_G(\boldsymbol{f})$ is accepted. Otherwise, it is rejected. The ratio $\tau_G(\boldsymbol{\xi})$ can be understood as a "signal-to-noise ratio". If $\boldsymbol{g}(\cdot) \in \vec{V}_G(\boldsymbol{f})$, the numerator is a squared noise norm and the denominator is the difference of the

"part" of the image $\boldsymbol{\xi}(\cdot)$ that can be an image of G from a uniformly bright image.

The ratio $\tau_G(\boldsymbol{\xi})$ is called morphological proximity of $\boldsymbol{\xi}(\cdot)$ to $\vec{V}_G(\boldsymbol{f})$ or shape proximity.

A practical example of the application of this method is shown in Figure 2.3, where the tracking of the clouds' fragment on video sequence is performed. Figures 2.3(a) and 2.3(b) show two gray-scale images $\boldsymbol{f}(\cdot)$ and $\boldsymbol{g}(\cdot)$ of the same scene G under different registration conditions. In image $\boldsymbol{f}(\cdot)$ in Figure 2.3(a), the subset H of the field of view X containing the selected fragment is shown. The problem is to find its position in the image $\boldsymbol{g}(\cdot)$.

To solve this problem, the shape $V_G(\boldsymbol{f}_H)$ of the selected fragment $\boldsymbol{f}_H$ of the image of G was constructed. The original fragment was approximated by a piecewise (mosaic) image. The class of its possible transformations modeling change of recording conditions was chosen as in Section 2.2.1. The subset H was moved over the field of view X, and for each position γH, the fragment $\boldsymbol{g}_{\gamma H}$ of the image $\boldsymbol{g}$ was compared to $V(\boldsymbol{f}_H)$ by calculating its morphological proximity

$$\tau(\boldsymbol{g}_{\gamma H}) = \frac{\|\boldsymbol{g}_{\gamma H} - \mathbf{P}_{\boldsymbol{f}_H}\boldsymbol{g}_{\gamma H}\|^2}{\|\mathbf{P}_0\boldsymbol{g}_{\gamma H} - \mathbf{P}_{\boldsymbol{f}_H}\boldsymbol{g}_{\gamma H}\|^2}. \tag{2.8}$$

The fragment $\boldsymbol{g}_{\gamma H}$ of the image $\boldsymbol{g}(\cdot)$ that is the closest to the given fragment $\boldsymbol{f}_H$ was chosen by minimizing $\tau(\boldsymbol{g}_{\gamma H})$ as a function of position γH. This found fragment is shown in Figure 2.3(b).



(a) (b)

Figure 2.3. Detection of similar segments in two images ((a) and (b)) of clouds obtained by two photo cameras under different registration conditions (see the areas inside the black rectangles).

2.2.4.2 *Morphological image filtering*

Morphological methods of classification have high noise immunity. Reference (noiseless and unblurred) images of digits, as well as a blurred image of the digit "1", are shown in Figure 2.4(a). The images of digits "1", "7", and "0" that are to be analyzed are shown in Figure 2.4(b). In this example, mosaic image shape was used. The field of view was split into sign area and background area. In addition, it was assumed that sign brightness is lower than background brightness. The algorithm that minimizes the belief Bel of an error (see Section 2.5.4) has recognized the images of "1" among the analyzed images without errors (see Section 9.8 in [35]). Figure 2.4(c) illustrates noise immunity on the images of the digit "seven" distorted by another type of noise, namely, circular spots of different diameters with non-uniform brightness that randomly varies between spots. The images are tabulated in the following way: each row is composed of images with the same spot radius, while images of each column have the same signal-to-noise ratio.

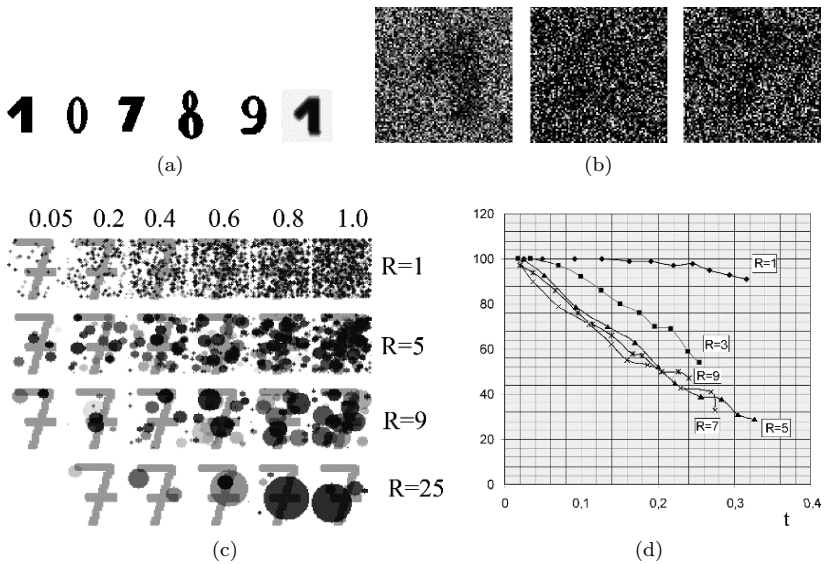


Figure 2.4. (a) Images of digits, with a blurred image of "1" on the right; (b) noisy blurred and arbitrarily shifted (within the field of view) images of digits that are presented for classification; (c) noisy images of the digit "seven"; (d) frequency of correct classification as a function of the signal-to-noise ratio.

In this example, mosaic image shape was used as well. The field of view was split into sign area and background area. In addition, it was assumed that sign brightness is lower than background brightness. Plots of the frequency of correct decisions as a function of the signal-to-noise ratio for different distortion types (for circle radii $R = 1, 3, 5, 7, 9$) are shown in Figure 2.4(d). One can see that the lower the circle radius, the higher the noise suppression.

2.2.4.3 *Scene difference highlighting*

For given images $\boldsymbol{f}(\cdot)$ and $\boldsymbol{g}(\cdot)$ of two scenes G_1 and G_2 obtained under unknown registration conditions, it is required to find the difference between scenes. If $\boldsymbol{P_f}$ is the projector onto the set V_{G_1} of images of the first scene, $\boldsymbol{P_f g}$ is the best approximation of $\boldsymbol{g}(\cdot)$ by images from V_{G_1}, so the image $\boldsymbol{g} - \boldsymbol{P_f g}$ is the difference between $\boldsymbol{g}(\cdot)$ and images of G_1.

This method is illustrated by the following example (see Figure 2.5). It is required to locate the car shown in the image $\boldsymbol{g}(\cdot)$ (Figure 2.5(b)) that is missing from the image $\boldsymbol{f}(\cdot)$ (Figure 2.5(a)). The shape of the moving image fragment $\boldsymbol{f}_H$ is constructed in the same way as in the previous example. Morphological analysis method allows us to highlight this object (see Figure 2.5(c)).

2.2.4.4 *Scene parameter estimation*

Let the image $\boldsymbol{\xi}(\cdot)$ be obtained according to equation (2.7), where the "ideal" image belongs to the shape of scene G_λ that depends on



(a) (b) (c)

Figure 2.5. Image $\boldsymbol{f}(\cdot)$ (a), image $\boldsymbol{g}(\cdot)$ of the same scene with an emerged object (b), image of an appearing object highlighted (c).

a parameter $\lambda \in \Lambda \subset R^m$. It is required to estimate λ using $\boldsymbol{\xi}(\cdot)$. To solve this problem, let us find the values of λ for which the image $\boldsymbol{\xi}(\cdot)$ is sufficiently close to $\vec{V}_{G_\lambda}$, i.e. such that

$$\tau_{G_\lambda}(\boldsymbol{\xi}) \stackrel{\text{def}}{=} \frac{\|\boldsymbol{\xi} - \mathbf{P}_\lambda \boldsymbol{\xi}\|^2}{\|\mathbf{P_0}\boldsymbol{\xi} - \mathbf{P}_\lambda \boldsymbol{\xi}\|^2} \leq c_0,$$

maximal error of λ is the solution of the problem

$$\max_{\lambda \in \Lambda(\xi)} \|\lambda - \hat{\lambda}\|^2 = \arg \inf_{\lambda' \in \Lambda} \max_{\lambda \in \Lambda(\xi)} \|\lambda - \lambda'\|^2,$$

where $\Lambda(\boldsymbol{\xi}) = \{\lambda \in \Lambda : \tau_{G_\lambda}(\boldsymbol{\xi}) \leq c_0\}$.

2.3 Rival Similarity of Image Shapes [46]

Traditionally, the methods of morphological image analysis were aimed at answering the question "does an image have a given shape?", i.e. in comparing images to image shapes. With that, little attention was paid to comparing shapes with each other until recently. In this section, we discuss the shape-to-shape comparison technique based on a concept of rival similarity of image shapes relying on orthogonal projections.

2.3.1 *Motivation*

Let's consider a problem of classifying objects by their images with two classes. Let all the images of the objects belonging to class number i taken under all feasible viewing conditions form an image shape $\bar{V}_i, i = 1, 2$. Given an image $\boldsymbol{f} \in \bar{V}_{j_*}$, where a value of index $j_* \in \{1, 2\}$ is unknown and needs to be predicted, one can solve the classification problem using the morphological method of minimizing distance from image $\boldsymbol{f}$ to the shapes $\bar{V}_i$. According to this method, a decision-maker decides that the object depicted in image $\boldsymbol{f}$ belongs to class j if

$$\text{argmin}_{i=1,2} \|\boldsymbol{f} - \overline{\boldsymbol{P}}_i \boldsymbol{f}\|^2 = j, \tag{2.9}$$

where $\overline{\boldsymbol{P}}_i$ is a projection onto shape $\bar{V}_i$.

However, when one solves a real-world problem, he usually knows the shapes $\bar{V}_i$ only approximately. That is, original image shapes $\bar{V}_i$

are not known and some other shapes V_i that "approximate" the original ones are given instead, $i = 1, 2$. In such a case, the following equation is practically used instead of equation (2.9):

$$\operatorname{argmin}_{i=1,2} \|\boldsymbol{f} - \boldsymbol{P}_i \boldsymbol{f}\|^2 = j, \tag{2.10}$$

where $\boldsymbol{P}_i$ is a projection onto image shape V_i.

Therefore, it is important to know if equation (2.10) is satisfied for any $\boldsymbol{f} \in \bar{V}_j$. If it is so, replacing the original shapes $\bar{V}_i$ with their approximate variants V_i does not affect the morphological method for classifying objects by their images. Otherwise, such a replacement (and, therefore, using equation (2.10) instead of equation (2.9)) can degrade the classification accuracy in an unpredictable manner. The results given in the following provide us with the answer to this question.

2.3.2 *Definition of rival similarity and its role in morphological image analysis*

Let image shapes V_1, V_2, and V be linear subspaces and $\boldsymbol{P}_1, \boldsymbol{P}_2, \boldsymbol{P}$ be orthogonal projections onto them. We say that *image shape V is h-similar to V_1 with respect to V_2* iff for any $\boldsymbol{f} \in V$

$$\|\boldsymbol{f} - \boldsymbol{P}_2 \boldsymbol{f}\|^2 - \|\boldsymbol{f} - \boldsymbol{P}_1 \boldsymbol{f}\|^2 \geqslant h \|\boldsymbol{f}\|^2, \tag{2.11}$$

where h is a numerical parameter from the unit interval $[0, 1]$. We say that *rival similarity of image shape V to image shapes V_1, V_2 is defined* iff V is h-similar to V_1 with respect to V_2 or vice versa for some $h \in [0, 1]$.

Let's apply the concept of rival similarity to the classification problem with the image shapes defined approximately which has been described in the motivation paragraph above. Suppose the following conditions are met:

- Image shape $\bar{V}_1$ is h_1-similar to V_1 with respect to V_2 for some $h_1 \in [0, 1]$.
- Image shape $\bar{V}_2$ is h_2-similar to V_2 with respect to V_1 for some $h_2 \in [0, 1]$.

- In such a case, for any $\boldsymbol{f}_1 \in \bar{V}_1$ and any $\boldsymbol{f}_2 \in \bar{V}_2$,

$$\|\boldsymbol{f}_1 - \mathbf{P}_2\boldsymbol{f}_1\|^2 - \|\boldsymbol{f}_1 - \mathbf{P}_1\boldsymbol{f}_1\|^2 \geqslant h_1\|\boldsymbol{f}_1\|^2,$$
$$\|\boldsymbol{f}_2 - \mathbf{P}_1\boldsymbol{f}_2\|^2 - \|\boldsymbol{f}_2 - \mathbf{P}_2\boldsymbol{f}_2\|^2 \geqslant h_2\|\boldsymbol{f}_2\|^2,$$

  which implies

$$\text{argmin}_{i=1,2}\|\boldsymbol{f}_j - \mathbf{P}_i\boldsymbol{f}_j\|^2 = j, \quad j = 1, 2,$$

compare to equation (2.10). That is, using shapes V_i instead of $\bar{V}_i$ does not affect the morphological method for classifying objects by their images.

2.3.3 *A numerical measure of rival similarity*

As one can see, parameter h introduced in equation (2.11) has not been involved in the considerations above. To achieve our conclusion that replacing the original image shapes with their "approximations" does not affect the morphological method for classifying objects by their images, we only need the following:

$$\|\boldsymbol{f}_1 - \boldsymbol{P}_2\boldsymbol{f}_1\|^2 \geqslant \|\boldsymbol{f}_1 - \boldsymbol{P}_1\boldsymbol{f}_1\|^2, \quad \|\boldsymbol{f}_2 - \boldsymbol{P}_1\boldsymbol{f}_2\|^2 \geqslant \|\boldsymbol{f}_2 - \boldsymbol{P}_2\boldsymbol{f}_2\|^2,$$

for any $\boldsymbol{f}_1 \in \bar{V}_1$ and any $\boldsymbol{f}_2 \in \bar{V}_2$. Thus, we only need rival similarities of $\bar{V}_1$ to V_1 with respect to V_2 and of $\bar{V}_2$ to V_2 with respect to V_1 for $h = 0$.

In this paragraph, we claim that a value of parameter h can be used to define a numerical measure of rival similarity of image shapes. Indeed, while inequality $\|\boldsymbol{f} - \boldsymbol{P}_2\boldsymbol{f}\|^2 \geqslant \|\boldsymbol{f} - \boldsymbol{P}_1\boldsymbol{f}\|^2$ means that an image $\boldsymbol{f} \in V$ is closer to V_1 that to V_2, the right-hand part of (2.11) specifies how much it is closer. That is why we define a numerical measure of rival similarity as follows:

$$R(V|V_1, V_2) = \inf\{\|\mathbf{P}_1\mathbf{P}\boldsymbol{g}\|^2 - \|\mathbf{P}_2\mathbf{P}\boldsymbol{g}\|^2, \|\mathbf{P}\boldsymbol{g}\| = 1\} \in [-1, 1],$$

where $\boldsymbol{P}$ is a projection onto image shape V. We have composed the expression under infimum considering the following facts: $\forall \boldsymbol{f} \in V$, $\exists \boldsymbol{g} : \boldsymbol{f} = \boldsymbol{P}\boldsymbol{g}$, and

$$\|\boldsymbol{f} - \mathbf{P}_2\boldsymbol{f}\|^2 - \|\boldsymbol{f} - \mathbf{P}_1\boldsymbol{f}\|^2 = \|\mathbf{P}_1\boldsymbol{f}\|^2 - \|\mathbf{P}_2\boldsymbol{f}\|^2.$$

Thus, $R(V|V_1, V_2)$ is the maximum value of h satisfying equation (2.11) for any image $\boldsymbol{f} \in V$. If $R(V|V_1, V_2) \geqslant 0$, then image shape

V is h-similar to V_1 with respect to V_2 for any $h \in [0, R(V|V_1, V_2)]$. Inversely, if V is h-similar to V_1 with respect to V_2 for some $h \in [0, 1]$, then $R(V|V_1, V_2) \geqslant h \geqslant 0$.

The following theorem defines a practical method for calculating a value of $R(V|V_1, V_2)$.

Theorem. *Let* $\lambda_1, \lambda_2, \ldots$ *be the eigenvalues of operator* $\boldsymbol{P}(\boldsymbol{P}_1 - \boldsymbol{P}_2)\boldsymbol{P}$ *enumerated such that* $\lambda_i \neq 0$ *for* $i = 1, \ldots,$ rank $\mathbf{P}(\boldsymbol{P}_1 - \boldsymbol{P}_2)\boldsymbol{P}$. *Then* $R(V|V_1, V_2) = \inf\{\lambda_i, i = 1, \ldots, \dim V\}$.

Note that in some cases image shapes V_1, V_2, and V have a common part $V_0 = V_1 \cap V_2 \cap V \neq \{0\}$. This implies for any image $\boldsymbol{g} \in V_0$

$$\mathbf{P}_1\boldsymbol{g} = \mathbf{P}_2\boldsymbol{g} = \boldsymbol{P}\boldsymbol{g} = \boldsymbol{g} \Rightarrow \|\mathbf{P}_1\mathbf{P}\boldsymbol{g}\|^2 - \|\mathbf{P}_2\mathbf{P}\boldsymbol{g}\|^2 = 0,$$

therefore, $R(V|V_1, V_2) \leqslant 0$. In such a case, to measure a degree of rival similarity numerically, it is convenient to work in a subspace $V_0^{\perp}$ and to replace images with their projections onto $V_0^{\perp}$ in all equations above.

Figure 2.6 demonstrates a concept of rival similarity on real-world images, and Figure 2.7 demonstrates it on synthetic data. As one can see, the mathematical definition of rival similarity and its numerical measure $R(\cdot|\cdot, \cdot)$ conform to an intuitive understanding of geometric shape similarity.

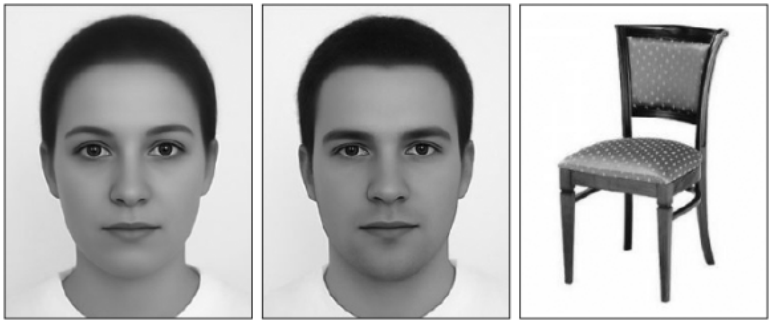


Figure 2.6. Demonstration of a concept of rival similarity on real-world images. The shape of the central image is similar to the shape of the left image with respect to the shape of the right one.

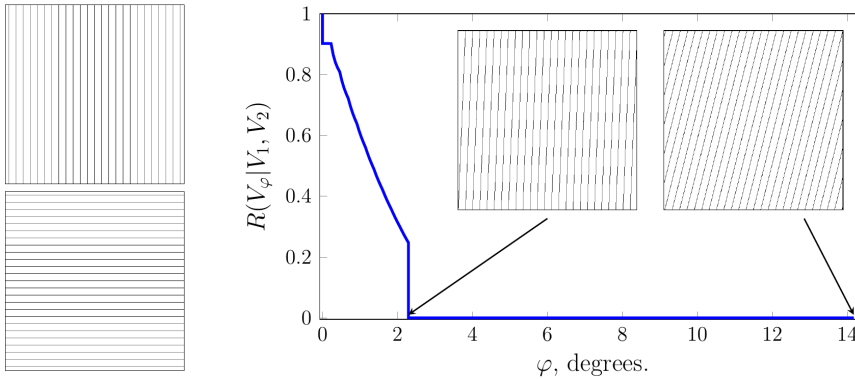


Figure 2.7. Demonstration of a numerical measure of rival similarity on synthetic data. Let V_1 and V_2 be the shapes of piecewise constant images consisting of vertical and horizontal bands, V_1 and V_1 are schematically shown in the upper-left corner and the lower-left one correspondingly. Let V_φ be the shape of a piecewise constant image consisting of inclined bands, the angle of inclination is φ. The graph of the numerical measure of rival similarity $R(V_\varphi|V_1, V_2)$ vs. φ is shown on the right-hand side. The calculations were performed in the orthogonal complement to the one-dimensional linear span V_0 of image χ_X which depicts an area of uniform brightness.

2.4 Oblique Projection in Image Morphology [33]

Let the Euclidean space $\vec{L}^2_\mu(X)$ (see Section 2.2) be decomposable into a direct but generally non-orthogonal sum of subspaces $\vec{L}^2_{\mu,1}(X)$ and $\vec{L}^2_{\mu,2}(X)$: $\vec{L}^2_\mu(X) = \vec{L}^2_{\mu,1}(X) \oplus \overleftarrow{L}^2_{\mu,2}(X)$. Therefore, any image $\boldsymbol{f}(\cdot) \in \vec{L}^2_\mu(X)$ can be uniquely represented as a sum of images $\boldsymbol{f}_1(\cdot) \in \vec{L}^2_\mu(X)$ and $\boldsymbol{f}_2(\cdot) \in \vec{L}^2_\mu(X)$: $\boldsymbol{f}(\cdot) = \boldsymbol{f}^{(1)}(\cdot) + \boldsymbol{f}^{(2)}(\cdot)$. The operator $S : \vec{L}^2_\mu(X) \to \overleftarrow{L}^2_\mu(X)$ defined by $S\boldsymbol{f}(\cdot) = \boldsymbol{f}^{(1)}(\cdot)$ is called the oblique projector that projects onto the space $\vec{L}^2_{\mu,1}(X)$ along $\vec{L}^2_{\mu,2}(X)$. In case of a finite-dimensional Euclidean space R^m for any subspace R^k there are infinitely many subspaces R^{m-k} complementing R^k, $R^m = R^k \oplus R^{m-k}$, and oblique projectors projecting onto R^k along R^{m-k} (see Figure 2.8).

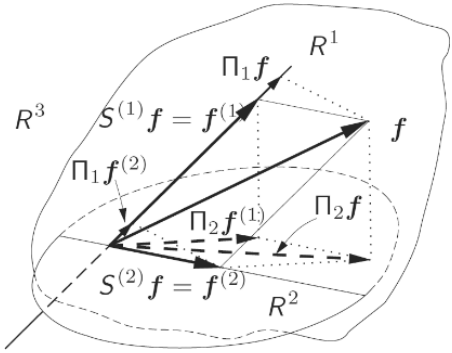


Figure 2.8. Orthogonal and oblique image projections.

2.4.1 *Shape as an oblique projector*

If in Figure 2.8 $R^1 = \{\alpha \boldsymbol{f}^{(1)}, \alpha \in (-\infty, +\infty)\}$ is the shape of the signal (image) $\boldsymbol{f}^{(1)}, R^2$ is the shape of the noise $\boldsymbol{f}^{(2)}, R^1 \cap R^2 = \{\mathbf{0}\}, R^1 \oplus R^2 = R^3$ is the shape of the class of observable signals $\boldsymbol{f} = \boldsymbol{f}^{(1)} + \boldsymbol{f}^{(2)}, \boldsymbol{f}^{(1)} \in R^1, \boldsymbol{f}^{(2)} \in R^2$, and Π_i orthogonally projects onto $R^i, i = 1, 2$, then *oblique projection* "extracts" from $\boldsymbol{f}$ the signal $\boldsymbol{f}^{(1)}$ and the noise $\boldsymbol{f}^{(2)}$: $S^{(1)}\boldsymbol{f} = \boldsymbol{f}^{(1)}, S^{(2)}\boldsymbol{f} = \boldsymbol{f}^{(2)}$ (if $\boldsymbol{f}^{(2)}$ is an image of interest like $\boldsymbol{f}^{(1)}$, then it "extracts" the images $\boldsymbol{f}^{(1)}$ and $\boldsymbol{f}^{(2)}$ from their sum $\boldsymbol{f} = \boldsymbol{f}^{(1)} + \boldsymbol{f}^{(2)}$). In case of orthogonal projection,

$$\Pi_1 \boldsymbol{f} = \Pi_1 \boldsymbol{f}^{(1)} + \Pi_1 \boldsymbol{f}^{(2)} = \boldsymbol{f}^{(1)} + \Pi_1 \boldsymbol{f}^{(2)}, \quad \Pi_2 \boldsymbol{f} = \Pi_2 \boldsymbol{f}^{(1)} + \boldsymbol{f}^{(2)} \tag{2.12}$$

are the most accurate in R^3 approximations of $\boldsymbol{f}(\cdot)$ by signal $\boldsymbol{f}^{(1)} \in R^1$ and noise $\boldsymbol{f}^{(2)} \in R^2$, and besides $\Pi_1 \boldsymbol{f} + \Pi_2 \boldsymbol{f} \neq \boldsymbol{f}$ (see example in Figure 2.9). What is more, if, e.g. $\boldsymbol{f}^{(1)} = 0$, then $\boldsymbol{f} = \boldsymbol{f}^{(2)}$ and according to equation (2.12) $\Pi_1 \boldsymbol{f} = \Pi_1 \boldsymbol{f}^{(2)}, \Pi_2 \boldsymbol{f} = \boldsymbol{f}^{(2)}$, so *instead of the signal* $S^{(1)}\boldsymbol{f} = \mathbf{0}$ *we obtain its most accurate approximation* $\Pi_1 \boldsymbol{f}^{(2)}$ *by the noise from* R^2, i.e. orthogonal projection does not allow to decompose $\boldsymbol{f}$ into the summands $\boldsymbol{f}^{(1)}$ and $\boldsymbol{f}^{(2)}$.

Orthogonal projectors Π_1 and Π_2 represent shapes of images $\boldsymbol{f}_1$ and $\boldsymbol{f}_2$ "irrespective of each other" in the sense that, for defining the projector Π_1 it is sufficient to know the shape R^1 and it is unimportant whether the projector Π_2 and the shape R^2 are defined. Therefore, *in the context of comparative analysis of image classes*, it is natural to call the shapes R^1, R^2 and the orthogonal projectors

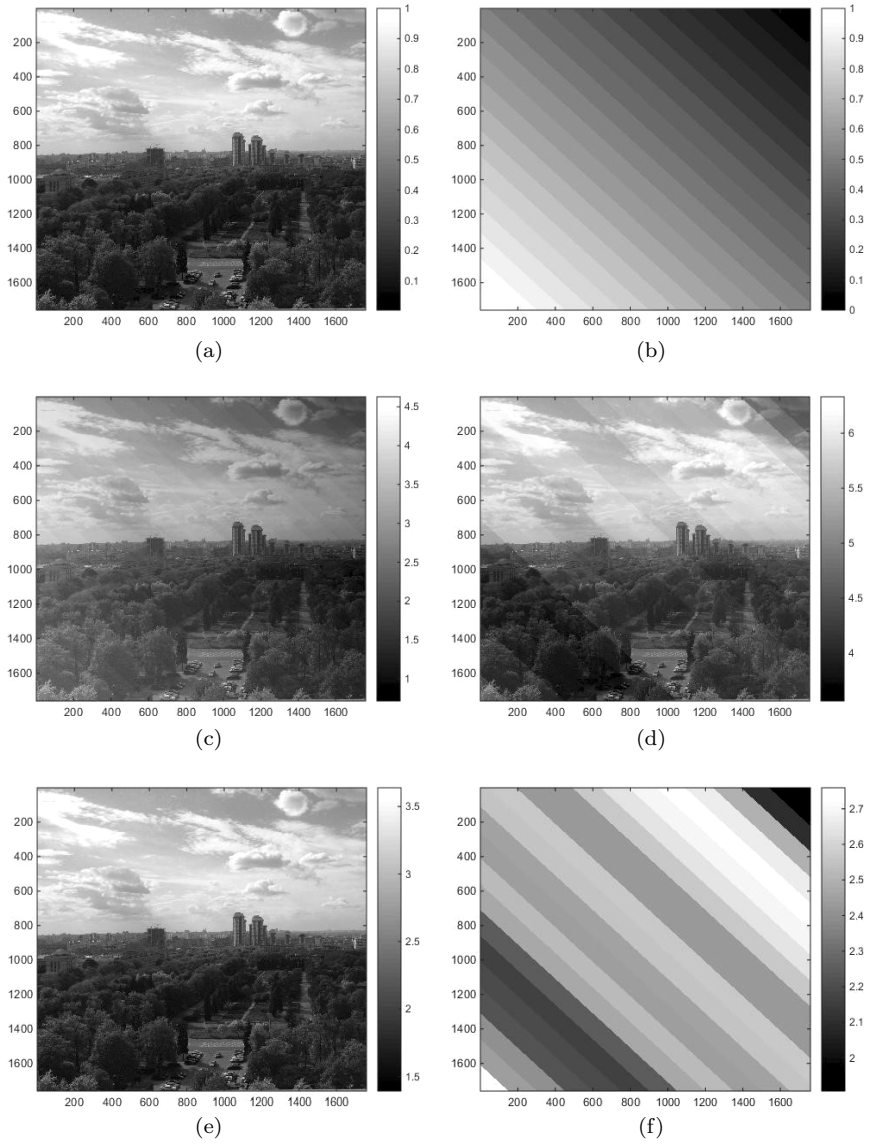


Figure 2.9. *The reference scene image* $\boldsymbol{f}_1$(a); *the image* $\boldsymbol{f}_2$*, whose field of view is split* into diagonal nonintersecting areas of constant brightness (b); the image $\boldsymbol{f} = 3\boldsymbol{f}_1 + 2\boldsymbol{f}_2$ that is to be analyzed (the shapes $V_{\boldsymbol{f}_1}$ and $V_{\boldsymbol{f}_2}$ are known) (c); an illustration of impossibility to decompose the image $\boldsymbol{f}$ using orthogonal projection of $\boldsymbol{f}$ onto the shapes $V_{\boldsymbol{f}_1}$ and $V_{\boldsymbol{f}_2}$: $\boldsymbol{f} \neq \Pi_1\boldsymbol{f} + \Pi_2 f$ (d); the image $\Pi_1\boldsymbol{f}$ (e); the image $\Pi_2\boldsymbol{f}$ (f). For oblique projection, one has $S^{(1)}\boldsymbol{f} = 3\boldsymbol{f}_1, S^{(2)}\boldsymbol{f} = 2\boldsymbol{f}_2$ and $\boldsymbol{f} = S^{(1)}\boldsymbol{f} + S^{(2)}\boldsymbol{f}$.

Π_1 and Π_2 representing them, the *absolute shapes*, and to call the oblique projectors $S^{(1)}$ and $S^{(2)}$ the *relative shapes*.

For applications of oblique projection in problems of comparative analysis of image class shapes, their morphological dependency, connectedness, determining the indices of morphological dependency and connectedness as prior characteristics of quality of solutions of morphological filtering problems, object image extraction, etc., see [33].

2.5 Subjective Modeling in Image Morphology [25, 34]

Mathematical methods and models of morphological analysis and interpretation of real scene images are, as a rule, software and mathematical implementations of results of subjective analysis by a researcher of the research objective and the means (mathematical, computational, etc.) of achieving it. They (usually!) allow formulating a *subjective morphological model of visual information about the research object that is contained in its images.*

2.5.1 *Uncertain element*

The mathematical formalism of subjective modeling was proposed in [32, 34] as a method of modeling incomplete and unreliable information about possible values of an unknown parameter $y \in Y$ determining a model $M(y)$ of the research object expressed by the researcher's subjective judgments. This allows us to model unformalized, incomplete and unreliable knowledge, ranging from "absolute ignorance" of the research object model to "exact knowledge" of this model. Furthermore, if fuzzy or random observation data of the research object is available, it is possible to empirically verify the adequacy of the subjective model for the research objective, to correct the subjective model, and under certain conditions to empirically recover the research object's fuzzy or random model.

The mathematical foundation of this formalism is a space $(Y, P(Y), \mathrm{Pl}^{\tilde{y}}, \mathrm{Bel}^{\tilde{y}})$ with plausibility $\mathrm{Pl}^{\tilde{y}} : P(Y) \to L$ and belief $\mathrm{Bel}^{\tilde{y}} : P(Y) \to \hat{L}$ measures *suggested by the researcher*, where Y is the set of all possible values of the unknown parameter y of the

model $M(y)$, $P(Y)$ is the class of all subsets of Y, and the *uncertain element* (u.el.) $\tilde{y}$ taking values in Y *characterizes modalities of researcher's subjective judgments about the truth of each value of* $y \in Y$ *by plausibility* $\mathrm{Pl}^{\tilde{y}}(\tilde{y} = y)$ *of equality* $\tilde{y} = y$ *and belief* $\mathrm{Bel}^{\tilde{y}}(\tilde{y} \neq y)$ *of inequality* $\tilde{y} \neq y$ [32].

Plausibility $\mathrm{Pl}^{\tilde{y}}$ and belief $\mathrm{Bel}^{\tilde{y}}$ take values respectively in scales $L = ([0,1], \leq, +, \times)$ and $\hat{L} = ([0,1], \geq, \hat{+}, \hat{\times})$, where addition $+$, $\hat{+}$ and multiplication $\times$, $\hat{\times}$ operations are defined according to

$$a + b = \max\{a, b\}, \quad a \,\hat{+}\, b = \min\{a, b\},$$
$$a \times b = \min\{a, b\}, \quad a \,\hat{\times}\, b = \max\{a, b\}, \quad a, b \in [0,1],$$

if they are continuous and commutative and satisfy certain conditions at boundaries of $[0,1]^2$. For any set $E \in P(X)$, measures $\mathrm{Pl}^{\tilde{y}}$ and $\mathrm{Bel}^{\tilde{y}}$ are defined according to

$$\mathrm{Pl}^{\tilde{x}}(E) = \sup_{x \in E} t^{\tilde{x}}(x), \quad E \neq \varnothing, \quad \mathrm{Pl}^{\tilde{x}}(\varnothing) = 0, \quad \mathrm{Pl}^{\tilde{x}}(X) = 1,$$

$$\mathrm{Bel}^{\tilde{x}}(E) = \inf_{x \in X \setminus E} \hat{t}^{\tilde{x}}(x), \quad E \neq \varnothing, \quad \mathrm{Bel}^{\tilde{x}}(X) = 1, \quad \mathrm{Bel}^{\tilde{x}}(\varnothing) = 0,$$

where $t^{\tilde{y}}(y) = \mathrm{Pl}^{\tilde{y}}(\tilde{y} = y)$, $\hat{t}^{\tilde{y}}(y) = \mathrm{Bel}^{\tilde{y}}(\tilde{y} \neq y)$, $y \in Y$. Functions $t^{\tilde{y}}(\cdot) : Y \to L$ and $\hat{t}^{\tilde{y}}(\cdot) : Y \to \hat{L}$ are called plausibility and belief distributions of values of y, $t^{\tilde{y}}(y)$ and $\hat{t}^{\tilde{y}}(y)$ are the plausibility of the truth of equality $\tilde{y} = y$ and the belief in the truth of inequality $\tilde{y} \neq y$, $y \in Y$, and values of $\mathrm{Pl}^{\tilde{y}}(E)$ and $\mathrm{Bel}^{\tilde{y}}(E)$ determine the plausibility and belief in the truth of containment $\tilde{y} \in E \in P(Y)$. *The space* $(Y, P(Y), \mathrm{Pl}^{\tilde{y}}, \mathrm{Bel}^{\tilde{y}})$ *characterizes the uncertain element* $\tilde{y}$ *and is called its model.*

An uncertain element can be interpreted as a model of researcher's subjective judgments about possible values of $y \in Y$ and their modalities that are founded on a set-theoretic representation of propositional logic [32, 34]. Given this representation, measures $\mathrm{Pl}^{\tilde{y}}(\cdot)$ and $\mathrm{Bel}^{\tilde{y}}(\cdot)$ should be interpreted as modal operators defined on the set of all propositions $P(Y)$ that take values in scales L and $\hat{L}$, respectively, plausibility $t^{\tilde{y}}(y) = \mathrm{Pl}^{\tilde{y}}(\tilde{y} = y)$ $(\mathrm{Pl}^{\tilde{y}}(\tilde{y} \in E))$ is interpreted as plausibility of the truth of the subjective judgment as the uncertain proposition that $\tilde{y} = y (\tilde{y} \in E)$, where $y \leftrightarrow \{y\}$ $(e \leftrightarrow E)$, belief $\hat{t}^{\tilde{y}}(y) = \mathrm{Bel}^{\tilde{y}}(\tilde{y} \neq y)$ $(\mathrm{Bel}^{\tilde{y}}(\tilde{y} \in E))$ is belief in the truth of the subjective judgment as the uncertain proposition $\tilde{y} \in Y \setminus \{y\}$ $(\tilde{y} \in E)$.

2.5.2 *Models of "absolute ignorance" and "exact knowledge" of the research object model*

The case of "absolute ignorance" of the model $M(y)$, $y \in Y$, is described by distributions $t^{\tilde{y}}(y) = \mathrm{Pl}^{\tilde{y}}(\tilde{y} = y) = 1$ and $\hat{t}^{\tilde{y}}(y) = \mathrm{Bel}^{\tilde{y}}(\tilde{y} \neq y) = 0$, $y \in Y$, that are independent from the choice of scales L and $\hat{L}$ and according to which all values of $\tilde{y} = y$ are equally plausible and any inequality $\tilde{y} \neq y$, $y \in Y$, cannot be believed in. In this case, *any consequence of the model* $M(y)$, $y \in Y$, *is absolutely unknown* [32].

In case of "exact knowledge" $t^{\tilde{y}}(y_0) = 1$, $t^{\tilde{y}}(y) = 0$ for $y \in Y \setminus_{\{y_0\}}$, $\hat{t}^{\tilde{y}}(y_0) =$, $\hat{t}^{\tilde{y}}(y) = 1$ for $y \in Y \setminus_{\{y_0\}}$, i.e. y_0 is the only plausible value of the parameter $y \in Y$ of the model $M(y)$ and any other value cannot be believed in. In this case, *any consequence of the model* $M(y)$ *is exactly known* [32]. These facts mean that subjective modeling is a generalization of "standard" mathematical modeling (see [25]).

2.5.3 *Subjective model and rules of estimation* [32]

We consider an *estimation* problem, in which the researcher defined a subjective model for the observation of the research object as a space $(Z \times K, P(Z \times K), \mathrm{Pl}^{\tilde{z},\tilde{k}}, \mathrm{Bel}^{\tilde{z},\tilde{k}})$, in which the measures $\mathrm{Pl}^{\tilde{z},\tilde{k}}$ and $\mathrm{Bel}^{\tilde{z},\tilde{k}}$ were specified by distributions $t^{\tilde{z},\tilde{k}}(z,k)$ and $\hat{t}^{\tilde{z},\tilde{k}}(z,k)$, $z \in Z$,$k \in K$, of a pair $\tilde{z},\tilde{k}$ of uncertain elements (observation, state), the first of which is observable and the second is not. The estimation problem requires using the observation $\tilde{z} = z$ to make a decision $d(z) \in K$ about the object state if the plausibility and belief of "losses" caused by assuming that the object is in the state $d(z)$ while its true state is k are equal to $\mathrm{pl}_{k,d}(z)$ and $\mathrm{bel}_{k,d}(z), z \in Z$.

The functions $\mathrm{pl}_{\cdot,\cdot}(\cdot) : K \times K \to L$ and $\mathrm{bel}_{\cdot,\cdot}(\cdot) : K \times K \to \hat{L}$ are defined by the decision-making subject (d.m.s.) who understands the consequences of his wrong decisions. For this purpose, he employs a *family* of spaces $(El, P(El), \mathrm{Pl}(\cdot|\cdot;d), \mathrm{Bel}(\cdot|\cdot;d))$, $d \in K$, in which El is the space of elementary "losses", the set $P(El)$ is the class of all subsets of El, and measures $\mathrm{Pl}(\cdot|\cdot;d) : P(El) \times T \to L$ and $\mathrm{Bel}(\cdot|\cdot;d) : P(El) \times T \to \hat{L}$ are the *transitional plausibility and belief* for $(K, P(K))$, $(El, P(El))$ for every $d \in K$. Further the d.m.s. singles out the set $V \in P(El)$ of elementary "losses" significant for him

and sets $\mathrm{pl}_{k,d} \stackrel{\mathrm{def}}{=} \mathrm{Pl}_{k,d}(V|k,d)$, $\mathrm{Bel}_{k,d} \stackrel{\mathrm{def}}{=} \mathrm{Bel}_{k,d}(V|k,d)$, $k,d \in K \times K$. The quality of estimation performed by the researcher and d.m.s. is determined by the *plausibility* and *belief*

$$\mathrm{Pl}(d(\cdot)) = \sup_{z\in Z}\max_{k\in K}\min(\mathrm{pl}_{k,d(z)}, t^{\tilde{z},\tilde{k}}(z,k)), \qquad (2.13)$$

$$\mathrm{Bel}(d(\cdot)) = \inf_{z\in Z}\min_{k\in K}\max(\mathrm{bel}_{k,d(z)}, \hat{t}^{\tilde{z},\tilde{k}}(z,k)), \qquad (2.14)$$

of the "losses", associated with the rule of the object state estimation specified by the decision function $d(\cdot) \in Z \to K$.

We define the functions $d^*(\cdot) \in Z \to K$ and $\hat{d}^*(\cdot) \in Z \to K$ as the solutions of the problems

$$\mathrm{Pl}_d(z) = \max_{k\in K}\min(\mathrm{pl}_{k,d}, t^{\tilde{z},\tilde{k}}(z,k)) \sim \min_{d\in K}, \qquad (2.15)$$

$$\mathrm{Bel}_d(z) = \min_{k\in K}\max(\mathrm{bel}_{k,d}, \hat{t}^{\tilde{z},\tilde{k}}(z,k)) \sim \min_{d\in K}, \qquad (2.16)$$

$z \in Z$. Any of such decision functions $d^*(\cdot) \in Z \to K$ and $\hat{d}^*(\cdot) \in Z \to K$ satisfies the conditions

$$d^*(\cdot) \in D(z) = \{d \in K, \mathrm{Pl}_d(z) = \min_{d'\in K}\mathrm{Pl}_{d'}(z)\}, \qquad (2.17)$$

$$\hat{d}^*(\cdot) \in \hat{D}(z) = \{d \in K, \mathrm{Bel}_d(z) = \min_{d'\in K}\mathrm{Bel}_{d'}(z)\}, \qquad (2.18)$$

$z \in Z$, and according to (2.13)–(2.18) as can easily be seen, they are the solutions of the problems

$$\min_{d(\cdot)\in K}\mathrm{Pl}(d(\cdot)) = \mathrm{Pl}(d^*(\cdot)) = \sup_{z\in Z}\mathrm{Pl}_{d^*(z)}(z),$$

$$\min_{d(\cdot)\in K}\mathrm{Bel}(d(\cdot)) = \mathrm{Bel}(\hat{d}^*(\cdot)) = \inf_{z\in Z}\mathrm{Bel}_{\hat{d}^*(z)}(z),$$

which define the optimum identification rules $d^*(\cdot) \in Z \to K$ and $\hat{d}^*(\cdot) \in Z \to K$, which minimizes the plausibility and belief of the "losses".

Note. Let

$$\mathrm{pl}_{k,d} = \begin{cases} 0, & \text{if } k = d, \\ 1, & \text{if } k \neq d, \end{cases}, \quad \mathrm{bel}_{k,d} = \begin{cases} 0, & \text{if } k = d, \\ > 0, & \text{if } k \neq d, \end{cases}, \quad k,d \in K \times K.$$

then problem (2.15) $\max_{k\in K}$ min $(\mathrm{pl}_{k,d}, t^{\tilde{z},\tilde{k}}(z,k)) \sim \min_{d\in K}$ is equivalent to the definition $d^*(z)$ from the condition $t^{\tilde{z},\tilde{k}}(z,d^*(z)) = \min_d\max_{k\neq d} t^{\tilde{z},\tilde{k}}(z,k)$, which is satisfied by each decision $d^*(\cdot)$ of the

problem $\max_k t^{\tilde{z},\tilde{k}}(z,k)$, which is called the *decision function of the maximum plausibility*. Problem (2.16) is equivalent to the problem $\hat{t}^{\tilde{z},\tilde{k}}(z,\hat{d}^*(z)) = \min_d \hat{t}^{\tilde{z},\tilde{k}}(z,k)$, $z \in Z$, while in the case of the dual consistency, $\hat{t}^{\tilde{z},\tilde{k}}(z,k) = \theta \circ t^{\tilde{z},\tilde{k}}(z,k)$, $z \in Z$, $k \in K$, of the measures $\mathrm{Pl}^{\tilde{z},\tilde{k}}$ and $\mathrm{Bel}^{\tilde{z},\tilde{k}}$ the set of its solutions coincides with the class of decision functions of the maximum plausibility.

2.5.4 *Subjective morphological model* [34]

As shown in [34], oblique projection is a particular case of solving the problem of *optimal estimation* of a signal $\boldsymbol{f}^{(1)}$ in a *subjective morphological observation model* defined by the researcher and in which the signal $\boldsymbol{f}^{(1)}$ and the noise $\boldsymbol{f}^{(2)}$ are the realizations of subjectively independent uncertain vectors $\tilde{\boldsymbol{f}}^{(1)}$ and $\tilde{\boldsymbol{f}}^{(2)}$, which are specified by the distributions

$$t^{\tilde{\boldsymbol{f}}^{(i)}}(\boldsymbol{f}^{(i)}) = \begin{cases} 1, \boldsymbol{f}^{(i)} \in R^i, \\ 0, \boldsymbol{f}^{(i)} \notin R^i, \end{cases} \quad i = 1,2, R^1 \oplus R^2 = R^3,$$

of the plausibilities of (the truth of) their values observed using the scheme $\tilde{\boldsymbol{f}} = \tilde{\boldsymbol{f}}^{(1)} + \tilde{\boldsymbol{f}}^{(2)}$ (it should be noted that in this case the shapes R^1 and R^2 are known but the distributions of plausibilities are given on R^1, R^2 according to the subjective model of "absolute ignorance"). Indeed, by virtue of the subjective independence of $\tilde{\boldsymbol{f}}^{(1)}$ and $\tilde{\boldsymbol{f}}^{(2)}$, their joint distribution $t^{\tilde{\boldsymbol{f}}^{(1)},\tilde{\boldsymbol{f}}^{(2)}}(\boldsymbol{f}^{(1)},\boldsymbol{f}^{(2)}) = \min\{t^{\tilde{\boldsymbol{f}}^{(1)}}(\boldsymbol{f}^{(1)}), t^{\tilde{\boldsymbol{f}}^{(2)}}(\boldsymbol{f}^{(2)})\}$, $\boldsymbol{f}^{(1)} \in R^1$, $\boldsymbol{f}^{(2)} \in R^2$, therefore,

$$t^{\tilde{\boldsymbol{f}}^{(1)},\tilde{\boldsymbol{f}}}(\boldsymbol{f}^{(1)},\boldsymbol{f}) = \min\{t^{\tilde{\boldsymbol{f}}^{(1)}}(\boldsymbol{f}^{(1)}), t^{\tilde{\boldsymbol{f}}^{(2)}}(\boldsymbol{f} - \boldsymbol{f}^{(1)})\}$$

is the plausibility of the equalities $\tilde{\boldsymbol{f}}^{(1)} = \boldsymbol{f}^{(1)}$, $\tilde{\boldsymbol{f}} = \boldsymbol{f}$, and the optimal estimate $\boldsymbol{f}_*^{(1)} = \boldsymbol{f}^{(1)}(\boldsymbol{f})$, minimizing the *plausibility of an error* in estimating the value $\tilde{\boldsymbol{f}}^{(1)} = \boldsymbol{f}^{(1)}$, is defined by the condition

$$t^{\boldsymbol{f}^{(1)},\tilde{\boldsymbol{f}}}(\boldsymbol{f}_*^{(1)},\boldsymbol{f}) = \max\{t^{\tilde{\boldsymbol{f}}^{(1)},\tilde{\boldsymbol{f}}}(\boldsymbol{f}^{(1)},\boldsymbol{f}) | \boldsymbol{f}^{(1)} \in R^1\},$$

according to which $\boldsymbol{f}_*^{(1)} \in R^1 \cap (R^2 + \boldsymbol{f}) = \{S^{(1)}\boldsymbol{f}\} = \{\boldsymbol{f}^{(1)}\}$, $\boldsymbol{f}_*^{(1)} = S^{(1)}\boldsymbol{f}$ (see Figure 2.9) [34].

It should be noted that in the subjective scale $\gamma_{\boldsymbol{f}} L$, in which the plausibility $\gamma_{\boldsymbol{f}} \circ t^{\tilde{\boldsymbol{f}}}(\boldsymbol{f}) = 1$ of the equality $\tilde{\boldsymbol{f}} = \boldsymbol{f}$, where $t^{\tilde{\boldsymbol{f}}}(\boldsymbol{f}) = \sup\{t^{\tilde{\boldsymbol{f}}^{(1)},\tilde{\boldsymbol{f}}^{(2)}}(\boldsymbol{f}^{(1)}, \boldsymbol{f}^{(2)}) | \boldsymbol{f}^{(1)} \in R^1, \boldsymbol{f}^{(2)} \in R^2, \boldsymbol{f}^{(1)} + \boldsymbol{f}^{(2)} = \boldsymbol{f}\}$, $\boldsymbol{f} \in R^3$, is the distribution of plausibilities for the values of the observed signal $\tilde{\boldsymbol{f}}$, the distribution

$$\gamma_{\boldsymbol{f}} \circ t^{\boldsymbol{f}_*^{(1)}|\tilde{\boldsymbol{f}}}(\boldsymbol{f}^{(1)}|\boldsymbol{f}) = \gamma_{\boldsymbol{f}} \circ t^{\boldsymbol{f}_*^{(1)},\tilde{\boldsymbol{f}}}(\boldsymbol{f}^{(1)}, \boldsymbol{f})$$

of the uncertain vector $\tilde{\boldsymbol{f}}^{(1)}$ is conditional, provided that $\tilde{\boldsymbol{f}} = \boldsymbol{f}$ is maximal at $\tilde{\boldsymbol{f}}^{(1)} = \boldsymbol{f}_*^{(1)}$, i.e. *the optimum estimate of the signal is the estimate with the maximum conditional, under* $\tilde{\boldsymbol{f}} = \boldsymbol{f}$, *plausibility of the equality* $\tilde{\boldsymbol{f}}^{(1)} = \boldsymbol{f}_*^{(1)}$.

In this case, we deal with a *subjective morphological model* (s.m.m.) specified by the researcher as a *subjective shape of the uncertain element* $(\tilde{\boldsymbol{f}}^{(1)}, \tilde{\boldsymbol{f}}^{(2)})$, canonical for the space $(R^1 \times R^2, P(R^1 \times R^2), \mathrm{Pl})$, in which $(R^1 \times R^2)$ is a shape of a pair of signals $(\boldsymbol{f}^{(1)}, \boldsymbol{f}^{(2)})$, $\boldsymbol{f}^{(1)} \in R^1$, $\boldsymbol{f}^{(2)} \in R^2$, $P(R^1 \times R^2)$ is the class of all subsets $R^1 \times R^2$, the measure of plausibility $\mathrm{Pl}^{\tilde{\boldsymbol{f}}^{(1)},\tilde{\boldsymbol{f}}^{(2)}}(\cdot) : P(R^1 \times R^2) \to L$ is defined by the plausibility distribution $t^{\tilde{\boldsymbol{f}}^{(1)},\tilde{\boldsymbol{f}}^{(2)}}(\cdot,\cdot) : R^1 \times R^2 \to L$, and it is required to estimate the uncertain vector $\tilde{\boldsymbol{f}}^{(1)}$ based on the observation of the uncertain vector $\tilde{\boldsymbol{f}} = \tilde{\boldsymbol{f}}^{(1)} + \tilde{\boldsymbol{f}}^{(2)}$ with the values in $R^1 \oplus R^2$.

As for the empirical verification of the adequacy of the s.m.m. for the purpose of the research (in this case, the purpose of the research is to estimate the signal $\tilde{\boldsymbol{f}}^{(1)} = \boldsymbol{f}^{(1)}$ using the observation of the value $\tilde{\boldsymbol{f}} = \boldsymbol{f}$), in this case under any observation result $\tilde{\boldsymbol{f}} = \boldsymbol{f}$, the subjective model cannot be rejected because for all $\boldsymbol{f} \in R^3 t^{\tilde{\boldsymbol{f}}}(\boldsymbol{f}) = \sup\{t^{\tilde{\boldsymbol{f}}^{(1)},\tilde{\boldsymbol{f}}^{(2)}}(\boldsymbol{f}^{(1)}, \boldsymbol{f}^{(2)}) \,|\, \boldsymbol{f}^{(1)} \in R^1, \boldsymbol{f}^{(2)} \in R^2, \boldsymbol{f}^{(1)} + \boldsymbol{f}^{(2)} = \boldsymbol{f}\} = 1$, i.e. all values of $\boldsymbol{f} \in R^3 = R^1 \oplus R^2$ are equally plausible.

If the subjective shape of the pair of uncertain vectors $\tilde{\boldsymbol{f}}^{(1)}, \tilde{\boldsymbol{f}}^{(2)}$ is defined by the r.-m. as a space with the plausibility $(R_{(1)} \times R_{(2)}, P(R_{(1)} \times R_{(2)}), \mathrm{Pl}^{\tilde{\boldsymbol{f}}^{(1)},\tilde{\boldsymbol{f}}^{(2)}})$, where $R_{(1)}$ and $R_{(2)}$ are different one-dimensional subspaces of R^3 and

$$t^{\tilde{\boldsymbol{f}}^{(i)}}(\boldsymbol{f}^{(i)}) = \begin{cases} 1, \boldsymbol{f}^{(i)} \in R_{(i)}, \\ 0, \boldsymbol{f}^{(i)} \notin R_{(i)}, \end{cases} \quad i = 1, 2,$$

uncertain vectors are subjectively independent, then the distribution of the uncertain vector $\tilde{\boldsymbol{f}} = \tilde{\boldsymbol{f}}^{(1)} + \tilde{\boldsymbol{f}}^{(2)}$ is as follows:

$$t^{\tilde{\boldsymbol{f}}}(\boldsymbol{f}) = \sup\{\min\{t^{\tilde{\boldsymbol{f}}^{(1)}}(\boldsymbol{f}^{(1)}), t^{\tilde{\boldsymbol{f}}^{(2)}}(\boldsymbol{f}^{(2)})\} |$$

$$|\boldsymbol{f}^{(1)} \in R^1, \boldsymbol{f}^{(2)} \in R^2, \boldsymbol{f}^{(1)} + \boldsymbol{f}^{(2)} = \boldsymbol{f}\} = \begin{cases} 1, \boldsymbol{f}^{(i)} \in R_{(1)} \oplus R_{(2)}, \\ 0, \boldsymbol{f}^{(i)} \notin R_{(1)} \oplus R_{(2)}, \end{cases}$$

and consequently, s.m.m. proposed by the r.-m. is not adequate for the purpose of the research if the observed uncertain vector $\tilde{\boldsymbol{f}} = \boldsymbol{f} \notin R_{(1)} \oplus R_{(2)}$. If, on the other hand, $\tilde{\boldsymbol{f}} = \boldsymbol{f} \in R_{(1)} \oplus R_{(2)}$, then the maximum plausibility estimate of the *value* $\tilde{\boldsymbol{f}}^{(1)} = \boldsymbol{f}^{(1)}$ is the oblique projection $S^{(1)}_{(1)}\boldsymbol{f} = \boldsymbol{f}^{(1)}$, where $S^{(1)}_{(1)} : R_{(1)} \oplus R_{(2)} \to R_{(1)} \oplus R_{(2)}$ is the oblique projector, which projects onto $R_{(1)}$ along $R_{(2)}$.

Oblique projection will define the solution for the problem of the optimum estimation of the value $\tilde{\boldsymbol{f}}^{(1)} = \boldsymbol{f}^{(1)}$ if the r.-m. defines the subjective shape of the pair of uncertain vectors $\tilde{\boldsymbol{f}}^{(1)}, \tilde{\boldsymbol{f}}^{(2)}$ as a space with the plausibility $\left(R_{(1)+} \times R_{(2)+}, P(R_{(1)+} \times R_{(2)+}), \mathrm{Pl}^{\tilde{\boldsymbol{f}}^{(1)}, \tilde{\boldsymbol{f}}^{(2)}}\right)$, in which $R_{(1)+} = \{\alpha \boldsymbol{f}^{(i)}, \alpha \in [0, +\infty)\}$ is the shape of $\boldsymbol{f}^{(i)}$ (a one-dimensional cone in R^3), $R_{(1)+} \oplus R_{(2)+}$ is the shape of $\boldsymbol{f} = \boldsymbol{f}^{(1)} + \boldsymbol{f}^{(2)}$ (two-dimensional cone in $R^2 = R_{(1)} \oplus R_{(2)}$ and in R^3),

$$t^{\tilde{\boldsymbol{f}}^{(i)}}(\boldsymbol{f}^{(i)}) = \begin{cases} 1, \boldsymbol{f}^{(i)} \in R_{(i)+}, \\ 0, \boldsymbol{f}^{(i)} \notin R_{(i)+}, \end{cases} \quad i = 1, 2,$$

is the distribution of uncertain vector $\tilde{\boldsymbol{f}}^{(i)}, i = 1, 2$, $t^{\tilde{\boldsymbol{f}}^{(1)}, \tilde{\boldsymbol{f}}^{(2)}}(\boldsymbol{f}^{(1)}, \boldsymbol{f}^{(2)}) = \min_{i=1,2} t^{\tilde{\boldsymbol{f}}^{(i)}}(\boldsymbol{f}^{(i)})$, $\boldsymbol{f}^{(i)} \in R_{(i)}, i = 1, 2$, is the distribution of $\mathrm{Pl}^{\tilde{\boldsymbol{f}}^{(1)}, \tilde{\boldsymbol{f}}^{(2)}}$. In this case, the subjective model will be inadequate if the observed uncertain vector $\tilde{\boldsymbol{f}} = \boldsymbol{f} \notin R_{(1)} \oplus R_{(2)}$, and if $\tilde{\boldsymbol{f}} = \tilde{\boldsymbol{f}}^{(1)} + \tilde{\boldsymbol{f}}^{(2)} \in R_{(1)} \oplus R_{(2)}$, the sought maximum plausibility estimate of the value $\tilde{\boldsymbol{f}}^{(1)} = \boldsymbol{f}^{(1)}$ is the oblique projection $S^{(1)}_{(1)}\boldsymbol{f} = \boldsymbol{f}^{(1)}_*$, where $S^{(1)}_{(1)} : R_{(1)} \oplus R_{(2)} \to R_{(1)} \oplus R_{(2)}$ projects in $R_{(1)} \oplus R_{(2)}$ onto $R_{(1)}$ along $R_{(2)}$. See [25] for other examples of subjective modeling in image morphology.

2.5.5 *Relative shapes as oblique projectors*

We define the class of the scene images by the equality

$$\boldsymbol{f}^{(1)}(x) = \sum_{i=1}^{n} \boldsymbol{f}_i^{(1)}(x)\chi_i(x), \quad x \in X, \boldsymbol{f}_i^{(1)}(\cdot) \in \boldsymbol{F}_i^{(1)}, \quad i = 1, \ldots, n, \tag{2.19}$$

in which $\chi_1(\cdot), \ldots, \chi_n(\cdot)$ are indicator functions (i.f.) of the subsets $A_1, \ldots, A_n$ of the field of vision X, which *symbolize* elements and form a partition $X = A_1 \cup A_2 \cup \cdots \cup A_n$; the unknown function $\boldsymbol{f}_i^{(1)}(\cdot) \in \boldsymbol{F}_i^{(1)}$ models *possible optical properties and lighting of i-th element of the* $G^{(1)}$ presented on the image (2.19), i.f. $\chi_i(\cdot)$, $i = 1, \ldots, n$, of the set

$$\boldsymbol{F}_i^{(1)} = \left\{\left(\boldsymbol{a}_{1i}^{(0)} + \boldsymbol{a}_{1i}^{(1)}x_1 + \boldsymbol{a}_{1i}^{(2)}x_2\right)\chi_i(x), x \stackrel{\text{def}}{=} (x_1, x_2) \in X, \boldsymbol{a}_{1i}^{(t)} \in R^N,\right.$$
$$\left. t = 0, 1, 2, \mu(A_i) > 0\right\}, \quad i = 1, \ldots, n, \tag{2.20}$$

are *mutually morphologically independent* shapes [33] of the images $\boldsymbol{f}_i^{(1)}(x)\chi_i(x)$, $x \in X, i = 1, \ldots, n$, of the scene elements represented on the images (2.19) by the indicator functions $\chi_i(\cdot)$, $i = 1, \ldots, n$ (mutually orthogonal subspaces $L_\mu^2(X)$).

We denote by $L^{(1)} = \oplus_{i=1}^{n} \boldsymbol{F}_i^{(1)}$ the minimum subspace[a] containing all images (2.19) of the scene $G^{(1)}$ ($L^{(1)} = V_{G^{(1)}}$ is the shape of the images $G^{(1)}$)$\Pi^{(1)} : L_\mu^2(X) \to L^{(1)}$ is the orthogonal projector on $L^{(1)}$ representing the shape $V_{G^{(1)}}$. The class of the images of the same scene, in which a certain object has appeared, optical properties of the elements of the scene $G^{(1)}$, and the conditions of their lightning have changed is defined by the equality

$$\boldsymbol{f}^{(2)}(x) = (1 - \chi_0(x)) \sum_{i=1}^{n} \boldsymbol{f}_i^{(2)}(x)\chi_i(x) + \boldsymbol{f}_0^{(2)}(x),$$
$$x \in X, \boldsymbol{f}_i^{(2)}(\cdot) \in \boldsymbol{F}_i^{(2)}, \quad i = 1, \ldots, n, \tag{2.21}$$

[a] To be more precise, $L = \mathrm{Cl}(L^{(1)} + L^{(2)})$, where Cl is the symbol of closure in $L_\mu^2(X)$, but further $L^{(1)}$, $L^{(2)}, \ldots$ are finite-dimensional subspaces of $L_\mu^2(X)$ and their finite sums are subspaces (closed linear manifolds) of $L_\mu^2(X)$.

in which $\chi_0(\cdot)$, the i.f. of the set $A_0 \subset X$ *symbolizing the emerged object*, represents it on the image (2.21), the unknown function

$$\boldsymbol{f}_0^{(2)}(\cdot) \in \boldsymbol{F}_0^{(2)} = \left\{\left(\boldsymbol{a}_0^{(0)} + \boldsymbol{a}_0^{(1)} x_1 + \boldsymbol{a}_0^{(2)} x_2\right) \chi_0(x), \quad x = (x_1, x_2) \in X, \right.$$
$$\left. \boldsymbol{a}_0^{(t)} \in R^N, \quad t = 0, 1, 2, \mu(A_i) > 0 \right\} \tag{2.22}$$

models its optical properties and the lighting conditions, unknown functions $\boldsymbol{f}_i^{(2)}(\cdot)$ are written as

$$\boldsymbol{f}_i^{(2)}(\cdot) \in \boldsymbol{F}_i^{(2)} = \left\{ \left(\boldsymbol{a}_{2i}^{(0)} + \boldsymbol{a}_{2i}^{(1)} x_1 + \boldsymbol{a}_{2i}^{(2)} x_2\right) \chi_i(x), \quad x = (x_1, x_2) \in X, \right.$$
$$\left. \boldsymbol{a}_{2i}^{(t)} \in R^N, \quad t = 0, 1, 2, \mu(A_i) > 0 \right\}, \quad i = 1, \ldots, n.$$

The modified scene $G^{(1)}$ is denoted by $G^{(2)}$, the minimum $L_\mu^2(X)$ containing all images (2.21) is designated by $L^{(2)} (= V_{G^{(2)}})$, and we denote the orthogonal projector on $L^{(2)}$ representing the shape of the images of the scene $G^{(2)}$ as $\Pi^{(2)} : L_\mu^2(X) \to L^{(2)}$.

We designate by $L = L^{(1)} + L^{(2)}$ the minimum subspace of $L_\mu^2(X)$ containing $L^{(1)}$ and $L^{(2)}$. Further, all projectors with the values in L are treated as operators from L to L, and we consider $L^{(0)} = L^{(1)} \cap L^{(2)}$ as a subspace of L that characterizes the *connectedness* of the shapes $L^{(1)}$ and $L^{(2)}$. We define $L^{(1,0)} = L^{(1)} \cap L^{(0)\perp}$ and $L^{(2,0)} = L^{(2)} \cap L^{(0)\perp}$, where $L^{(0)\perp}$ is the orthogonal complement of $L^{(0)}$ in L.

Since $L = L^{(1,0)} \oplus L^{(0)} \oplus L^{(2,0)}$, as a result for all $\boldsymbol{f} \in L$,

$$\boldsymbol{f}(\cdot) = \underbrace{S^{(0)}\boldsymbol{f}}_{\in L^{(0)}} + \underbrace{S^{(1,0)}\boldsymbol{f}}_{\in L^{(1,0)}} + \underbrace{S^{(2,0)}\boldsymbol{f}}_{\in L^{(2,0)}}$$

$S^{(0)} : L \to L^{(0)}$ orthogonally projects onto $L^{(0)}$ along $L^{(1,0)} \oplus L^{(2,0)}$, $S^{(1)} : L \to L^{(1,0)}$ projects onto $L^{(1,0)}$ along $L^{(2,0)} \oplus L^{(0)} = L^{(2)}$, $S^{(2)} : L \to L^{(2,0)}$ projects onto $L^{(2,0)}$ along $L^{(1,0)} \oplus L^{(0)} = L^{(1)}$, where $S^{(0)} = \Pi^{(0)}$ is orthogonal since $L^{(1,0)} \oplus L^{(2,0)} \perp L^{(0)}$. When such a choice of $L^{(1,0)}$ and $L^{(2,0)}$ is made:

- $L^{(0)}$ and the projector $S^{(0)}$ define the *relative shape* of the image class of the elements common for the scenes and which are interpreted as *images of the virtual scene* $G^{(1)} \cap G^{(2)}$,

- $L^{(1,0)}$ and the projector $S^{(1,0)}$ define the relative shape of the image class of the virtual scene $G^{(1)}\backslash G^{(2)}$ as the images of the elements that are present in the scene $G^{(1)}$ and absent on $G^{(2)}$,
- $L^{(2,0)}$ and the projector $S^{(2,0)}$ define the relative shape of the image class of the virtual scene $G^{(2)}\backslash G^{(1)}$ as the images of the elements that are present in the scene $G^{(2)}$ and absent on $G^{(1)}$; $S^{(2,0)}$ represents the shape of the image class of the object *emerging* on $G^{(2)}$ and elements of $G^{(1)}$ distorted by its appearance.

The orthogonal projectors $\Pi^{(1)}$ and $\Pi^{(2)}$ represent the image shapes of the scenes $G^{(1)}$ and $G^{(2)}$ "irrespective" of each other in the sense that for defining $\Pi^{(1)}$, it is sufficient only to know $G^{(1)}$, therefore, the shapes $G^{(1)}$, $G^{(2)}$ and projectors $\Pi^{(1)}$, $\Pi^{(2)}$ representing them in [33] are referred to as the *absolute shapes*. The absolute shapes should be used in problems of signal detection in noise, the recognition of images in the noise, etc. when the alternative shape (the shape of the noise) *can be arbitrary*.

2.5.6 *Nonlocal method for object extraction*

In the morphological model (2.19)–(2.22), the shapes of the images of the scenes $G^{(1)}$ and $G^{(2)}$ are specified, and, in particular, we consider known both the domain A_0 of the field of vision symbolizing the object appearing on $G^{(2)}$ and the objects on the scene $G^{(1)}$ distorted by its appearance, which are extracted on the scene $G^{(2)}$ by the condition $\mu(A_0 \cap A_i) > 0$ for some $i \in \{1, \ldots, n\}$. Since the problem of extracting emerging and usually unknown objects is one of the basic problems in the morphological image analysis [35], we consider a new non-local morphological method for extracting the set A_0 and image $\boldsymbol{f}_0^{(2)}(x)\,\chi_0(x)$, $x \in X$, based on the model (2.19)–(2.22), in which for simplicity $\boldsymbol{a}_{1i}^{(t)} = \boldsymbol{a}_{2i}^{(t)} = 0$, $t = 1, 2$, $\boldsymbol{a}_{1i}^{(0)}, \boldsymbol{a}_{2i}^{(0)} \in \mathrm{R}^1$, $i = 1, \ldots, n$, $\boldsymbol{f}_0^{(2)}(x) = \chi_0(x) f(x)$, where $f(x)$ is an arbitrary function characterizing the optical properties of the emerged object. Thus, the "simplified" scenes $G^{(1)}$ and $G^{(2)}$ are denoted by $G_0^{(1)}$ and $G_0^{(2)}$. In this case, the shape of the image (2.19), (2.20) of the scene $G_0^{(1)}$ can be specified by algebra $\mathcal{A}$ of the subsets of the field of vision X generated by the partition $X = \bigcup_{i=1}^{n} A_i$.

We denote by $\mathcal{B}_m$ the inclusion-wise minimal algebra of the μ-measurable subsets of X containing $A_1, \ldots, A_n$ and the sets $B_{m,1}, \ldots, B_{m,k(m)}$, similar to $A_1, \ldots, A_n$ forming an μ-measurable partition $X = \bigcup_{k=1}^{k(m)} B_{m,k}$, $B_{m,s} \cap B_{m,t} = \varnothing$, $s \neq, t, s, t = 1, \ldots, k(m)$, and consider the sequence of algebras $\mathcal{B}_1 \subset \mathcal{B}_2 \subset \cdots$, in which the sets $B_{m+1,1}, \ldots, B_{m+1,k(m+1)}$ result from $B_{m,1}, \ldots, B_{m,k(m)}$ by the partition of some $B_{m,s}, s = 1, \ldots, k(m)$ into μ-measurable subspaces of the positive measure, $m = 1, 2, \ldots$.

It should be noted that for any B_m-measurable image $\boldsymbol{f}(\cdot) \in L^2_\mu(X)$, its orthogonal projection onto the class of $\mathcal{B}_m$-measurable images is

$$(\Pi_m^{(1)} \boldsymbol{f})(x) \stackrel{\text{def}}{=}$$

$$= \sum_{k=1}^{k(m)} \sum_{i=1}^{n} \frac{\chi_{m,k}(x)\chi_i(x)}{\|\chi_{m,k}\chi_i\|^2_\infty} \int_\chi \chi_{m,k}(x')\chi_i(x')\boldsymbol{f}(x')\mu(dx') = \boldsymbol{f}(x)$$

for μ-almost all $x \in X$ where in summation we do not take into account the terms in which

$$\|\chi_{m,k}\chi_i\|^2_\infty = 0, \quad k \in \{1, \ldots, k(m)\}, \quad i \in \{1, \ldots, n\},$$

and if $\mathcal{B}$ is the minimal σ-algebra containing $\mathcal{B}_{m,k}$, $k \in \{1, \ldots, k(m)\}$, $m = 1, 2, \ldots$, then for any $\mathcal{B}$-measurable image $\boldsymbol{f}(\cdot)$, one has

$$(\Pi_m^{(1)} \boldsymbol{f})(x) \underset{m\to\infty}{\longrightarrow} \boldsymbol{f}(x)$$

for μ-almost all $x \in X$. Moreover, for such images for μ-almost all

$$\lim_{m\to\infty} \frac{1}{\mu(B_m(x))} \int_{B_m(x)} \boldsymbol{f}(x')\mu(dx') = \boldsymbol{f}(x),$$

where $B_m(x) = B_{m,s(x)} \cap A_{i(x)}$ is a set with positive measure containing point x.

Let

$$B^0_{m,k} = \begin{cases} B_{m,k}, & \text{if } B_{m,k} \cap A_0 = \varnothing, \\ \varnothing, & \text{if } B_{m,k} \cap A_0 \neq \varnothing, \end{cases} \quad B^0_m = \bigcup_{k=1}^{k(m)} B^0_{m,k}$$

be a domain of the field of vision which does not contain A_0, then

- for any image $\boldsymbol{f}^{(2)}(\cdot)$ of the scene $G_0^{(2)}$ the image $(\Pi_m^{(1)}\boldsymbol{f}^{(2)})(x) - \boldsymbol{f}^{(2)}(x) = 0$ for μ-almost all $x \in B_m^0$ except for the points in the field of vision near the outer boundary of A_0, containing elements that are not $\hat{B}_m^0$-measurable defined by the boundary of A_0, therefore, the (observed) estimate of B_m^0 should be assumed to be the domain $\mathcal{B}_m^0 = \{x \in X, (\Pi_m^{(1)}\boldsymbol{f}^{(2)})(x) = \boldsymbol{f}^{(2)}(x)\}$, containing the $\mathcal{B}_m^0$-measurable part of $X\backslash A_0$ ($X\backslash A_0$ without the points near the outer boundary of A_0 that belong to $\mathcal{B}_0$-immeasurable sets defined by the border of A_0), and, possibly, a $\mathcal{B}_m^0$-measurable domain that is contained in A_0 (if $f(x) = \text{const}$, $x \in X$, then only the domain near the inner boundary of A_0 can be $\mathcal{B}_m^0$-immeasurable),
- as an estimate of the domain $X\backslash B_m^0$ containing the sought A_0, which decreases monotonically with increasing m because $B_m^0 \subset B_{m+1}^0$, we choose the domain $X\backslash\hat{B}_m^0$, $\mathcal{B}_m^0$-immeasurable sets inside and outside of the boundary of A_0, and
- we choose the image $\chi_{X\backslash\hat{B}_m^0}(x)\boldsymbol{f}^{(2)}(x)$, $x \in X$, as an estimate of the image of the emerged object and its neighborhood that is decreasing with increasing m.

If some morphological characteristics of the emerging object are known *a priori*, they should be taken into account when constructing $B_{m,k}$, $k = 1, \ldots, k(m)$, $m = 1, 2, \ldots$. The above nonlocal method for objects' extraction [34] is shown in Figure 2.10.

2.6 Conclusions

This chapter presents and investigates new efficient methods of morphological analysis. They are based on orthogonal and oblique projection in the Euclidean space of images, as well as on the mathematical formalism of subjective modeling that allows the researcher to mathematically formulate a subjective morphological model of the research object as a subjective model of the visual information about it that is contained in its images. The mathematical concepts of absolute and relative image shape, operations of image comparison in shape, and comparison of their shapes themselves are defined and investigated. Some new optimization problems of morphological image analysis are formulated and solved. The obtained results are illustrated with examples.

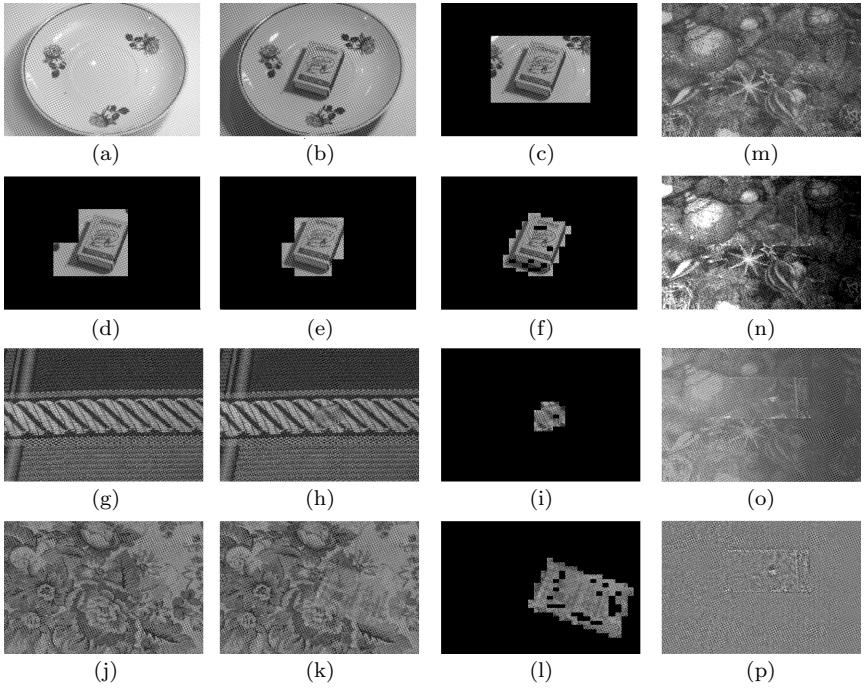


Figure 2.10. The results obtained by the algorithm for the extraction of the emerging object in real images. The "evolution" of the result depending on the partition refinement: (a) the reference image of the scene; (b) the image presented for analysis (the lighting conditions have changes or a new object has emerged); the results obtained by the algorithm for the partition of the field of vision: 4×4 px (c), 8×8 px (d), 16×16 px (e), 32×32 px (f). Extraction of semi-transparent objects: reference images (g, j); images presented for px analysis (h, k); results obtained by the partition 32×32 px (i, l). Comparison with image differences of (m) the reference image; (n) the image to px be analyzed (under different lighting and with an emerged "semi-transparent" object): (o) the difference of reference and analyzed images (that contains images of both the object and the background); (p) algorithm output for 39×26 px partition.

The morphological image analysis methods proposed in this chapter have well-grounded theoretical properties. Thus, they organically complement neural network methods that are popular at present and demonstrate high efficiency in applications but do not allow comprehensive theoretical analysis and, therefore, do not allow full interpretation and explanation of their operation. In the future, the ideas of morphological analysis and, in particular, the ideas of

subjective modeling of visual information can be combined with neural network approaches to give rise to new neural network architectures and training methods that will be transparent and human interpretable.

Acknowledgments

The authors are grateful to Yu. M. Nagorny and D. V. Golovin for the help in production of illustrations.

References

[1] Bai, X. (2007) Heat kernel analysis on graphs, PhD thesis, The University of York.

[2] Belkin, M. and Niyogi, P. (2001) Laplacian eigenmaps and spectral techniques for embedding and clustering. *Advances in Neural Information Processing Systems*, 14, 585–591.

[3] Brianskiy, S.A., Sidyakin, S.V., and Vizilter, Y.V. (2015) Orientation spectrum algorithm development, *International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, XL-5/W6, 13–18.

[4] Coifman, R. and Lafon, S. (2006) Diffusion maps. *Applied and Computational Harmonic Analysis*, 21(1), 5–30.

[5] Coifman, R.R., Lafon, S., Maggioni, M., Keller, Y., Szlam, A.D., Warner, F., and Zucker, S. (2006) Geometries of sensor outputs, inference and information processing. *Intelligent Integrated Microsystems*, 6232, 623209.

[6] De Goes, F., Goldenstein, S., and Velho, L. (2008) A hierarchical segmentation of articulated bodies. *Computer Graphics Forum*, 27(5), 1349–1356.

[7] Dougherty, E.R. (1989) The dual representation of gray-scale morphological filters. *Proceedings CVPR IEEE* (IEEE), pp. 172–177.

[8] Dubois, D. and Prade, H. (1988) *Possibility Theory. An Approach to Computerized Processing of Uncertainty* (Springer-Verlag, Berlin).

[9] Falomkina O.V., Pyt'ev, Y.P., Pyatkov, Y.V., and Kamanin, D.V. (2016) Estimation of reliability of linear point structures revealed in two-dimensional distributions of experimental data. *Journal of Physics: Conference Series,* 675, 042001.

[10] Falomkina O.V., Pyt'ev, Y.P., Pyatkov, Y.V., Kamanin, D.V., and Herbst, B.M. (2017) Oblique projectors in image morphology. *Journal of Physics: Conference Series,* 798(1), 012125.

[11] Falomkina O.V., Pyt'ev, Y.P., Pyatkov, Y.V., Kamanin, D.V., Herbst B.M., and Trzaska W.H. (2011) Estimating the reliability of revealing the fine structure in mass-kinetic energy distributions of nuclear reaction products. *Pattern Recognition and Image Analysis*, 21(1), 82–87.

[12] Gonzalez, R.C. and Woods, R.E. (2002) *Digital Image Processing*, 2nd Ed. (Prentice-Hall, Hoboken, NJ).

[13] Gori, M., Maggini, M., and Sarti, L. (2005) Exact and approximate graph matching using random walks. *IEEE Transactions on Pattern Analysis and Machine Intelligence,* 27(7), 1100–1111.

[14] Hough, P.V.C. (1962) Method and means for recognizing complex patterns, US Patent 3 069 654. 1962 December 18.

[15] Huang, T. (1975) *Picture Processing and Digital Filtering* (Springer-Verlag, Berlin).

[16] Hussain, M., Chen, D., Cheng, A., Wei, H., and Stanley, D. (2013) Change detection from remotely sensed images: From pixel-based to object-based approaches. *ISPRS Journal of Photogrammetry and Remote Sensing,* 80, 91–106.

[17] La, S., Chandra, M., Upadhyay, G.K., and Gupta, D. (2011) Removal of additive Gaussian noise by complex double density dual tree discrete wavelet transform. *MIT International Journal of Electronics and Communication Engineering*, 1(1), 8–16.

[18] Lafon, S. (2004) Diffusion maps and geometric harmonics, PhD thesis, Yale University, Department of Mathematics & Applied Mathematics.

[19] Lafon, S., Keller, Y., and Coifman, R.R. (2006) Data fusion and multicue data matching by diffusion maps. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 28(11), 1784–1797.

[20] Maragos, P. (1989) Pattern spectrum, multiscale shape representation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 11(7), 701–716.

[21] Mestetskiy, L. (2010) Skeleton representation based on compound Bezier curves. *VISAPP'2010, 5th International Conference on Computer Vision Theory and Applications*, Vol. 1 (INSTICC Press), pp. 44–51.

[22] Mestetskiy, L. and Semenov, A. (2008) Binary image skeleton-continuous approach. *VISAPP'2008, 3rd International Conference on Computer Vision Theory and Applications*, Vol. 1 (INSTICC Press), pp. 251–258.

[23] Mohideen, S.K. (2012) Denoising of images using complex wavelet transform. *International Journal of Advanced Science and Research,* 1(2), 176–184.

[24] Najman, L. and Talbot, H. (eds.) (2010) *Mathematical Morphology: From Theory to Applications* (ISTE Ltd, London and John Wiley & Sons, Inc., Hoboken, NJ).

[25] Pyt'ev, Y.P. (2018) Probability, possibility and subjective modeling in scientific research (Fizmatlit, Moscow) (in Russian).

[26] Pyt'ev, Y.P. (1975) Morphological concepts in image analysis problems. *Doklady Akademii Nauk SSSR,* 224, 1283–1286 (in Russian).

[27] Pyt'ev, Y.P. (1983) Morphological image analysis. *Doklady Akademii Nauk SSSR*, 269, 1061–1064 (in Russian).

[28] Pyt'ev, Y.P. (1984) Chapter: "Problems in morphological image analysis". *Mathematical Methods for the Earth's Natural Resource Exploration from Space* (Nauka, Moscow) (in Russian).

[29] Pyt'ev, Y.P. (1993) Morphological image analysis. *Pattern Recognition and Image Analysis*, 3, 19–28.

[30] Pyt'ev, Y.P. (1997) The morphology of color (multispectral) images. *Pattern Recognition and Image Analysis*, 7, 467–473.

[31] Pyt'ev, Y.P. (1998) Methods of morphological analysis of color images. *Pattern Recognition and Image Analysis*, 8, 517–531.

[32] Pyt'ev, Y.P. (2013) Modeling of subjective judgments made by a researcher-modeler about the model of the research object. *Mathematical Modeling and Computer Simulations*, 5(6), 538–557.

[33] Pyt'ev, Y.P. (2013) Oblique projectors and relative forms in image morphology. *Computational Mathematics and Mathematical Physics*, 53(12), 1916–1937.

[34] Pyt'ev, Y.P. (2017) Subjective models, oblique projectors, and optimal decisions in image morphology. *Pattern Recognition and Image Analysis,* 27(2), 213–233.

[35] Pyt'ev, Y.P. and Chulichkov, A.I. (2010) Methods of morphological image analysis (Fizmatlit, Moscow) (in Russian).

[36] Pyt'ev, Y.P. and Zhivotnikov, G.S. (2004) On the methods of possibility theory for morphological image analysis. *Pattern Recognition and Image Analysis*, 14, 60–71.

[37] Pyt'ev, Y.P., Kalinin, A.V., Loginov, E.O., and Smolovik, V.V. (1998) On the problem of object detection by black-and-white and color morphologies. *Pattern Recognition and Image Analysis*, 8, 532–536.

[38] Serra, J. (1982) *Image Analysis and Mathematical Morphology* (Academic Press, London).

[39] Sun, J., Ovsjanikov, M., and Guibas, L. (2009) A concise and provably informative multi-scale signature based on heat diffusion. *Computer Graphics Forum*, 28(5), 1383–1392.

[40] Urbach, E.R., Roerdink, J.B., and Wilkinson, M.H. (2007) Connected shape-size pattern spectra for rotation and scale-invariant classification of gray-scale images. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 29(2), 272–285.

[41] Vizilter, Y.V., Gorbatsevich, V.S., Rubis, A.Y., and Zheltov, S.Y. (2014) Shape-based image matching using heat kernels and diffusion maps. *International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, XL-3, 357–364.

[42] Vizilter, Y.V., Rubis, A.Y., and Gorbatsevich, V.S. (2012) Relational models of image shapes and shape comparison metrics. *Proceedings of 9th International Conference on "Intelligent Information Processing"*, pp. 410–414.

[43] Wilkinson, M. (2002) Generalized pattern spectra sensitive to spatial information. *Proceedings of 16th International Conference on Pattern Recognition*, Vol. 1, pp. 21–24.

[44] Zadeh, L.A. (1965) Fuzzy sets. *Information and Control,* 8, 235–350.

[45] Zhivotnikov, G.S., Pyt'ev, Y.P., and Falomkin, I.I. (2005) On the filtering algorithm for piecewise-constant images. *Intelligent Systems*, 9(1–4), 157–183.

[46] Zubyuk, A.V. (2017) Morphological methods for identifying objects of random shape by their images, PhD thesis, The Lomonosov Moscow State University.

© 2025 World Scientific Publishing Company
https://doi.org/10.1142/9789811267215_0003

Chapter 3

Image Analysis and Processing Based on Two-Dimensional Variations

Pavel A. Chochia* **and Olga P. Milukova†**

Institute for Information Transmission Problems of Russian Academy of Sciences, Moscow, the Russian Federation

**chochia@iitp.ru*

†milukova@iitp.ru

This chapter considers image analysis and processing problems based on a mathematical model of the signal and two-dimensional variations. A two-scale image model is formulated, and the applications of two-dimensional variations to discrete signals and image analysis are stated. The ways of evaluating the image complexity with the help of variations are explored. Changes in complexity estimates for different image transformations are analyzed. The issues of video signal separation into several information components are considered. An algorithm for image decomposition and extracting the component, which carries the basic information about the image objects, is given. An approach to formatting the topological characteristics of two-dimensional signals is proposed, and their use for estimating image parameters is considered. The index of object size and amplitude of convexity is introduced; their application for detecting the noise and objects of different sizes is demonstrated. During the formation, an image could be subjected to various distortions. The way to remove distortions is to solve the inverse problem. The most important task is the determination of the type and parameters of the distortion process. The task of diagnosing the distortion operator by analyzing the obtained image is discussed.

Keywords: Image analysis, image processing, image model, two-dimensional variations, image complexity, image decomposition, image smoothing, image components, color image processing, topological characteristics, object detection, distortion diagnostic, image restoration.

3.1 Introduction

Most of the video information analysis and processing problems require the evaluation of the obtained result; this question is important in research as in the practical application of interesting methods. In signal filtration and reconstruction tasks, some deviation estimates are usually used (MSE, PSNR, SSIM, *et al.*). This approach is suitable for theoretical research and other cases when the source signal is known. However, in real situations, undistorted source signal is inaccessible. Under these conditions, we need to use some solitary evaluation of the resulting signal, which is adequate for an image as a two-dimensional function.

There are two approaches to deciding this task. The most often approach consists of an attempt to estimate in one or another way the so-called "image quality" [26,50,51,61–63]. The second one concerns the introduction of some specific metric for evaluating the image complexity [44,64]. Every one of them has its own limitations.

Our task is to discover an universal estimate, which characterizes spatial two-dimensional changeability of image detail. The approach founded on the consideration of an image as a function of limited total variation [38] and estimation of its variability with two-dimensional variations mechanism seems to be productive [1,23,29,59]. At that time, the definition of changeability estimation is similar to the image complexity [38]. Two-dimensional Kronrod variation [29] looks an attractive way for the decision of this task.

We investigate the estimation of image complexity and interpret it as a characteristic reflecting the number, the sizes, and the contrasts of image details [14]. For this task, we use the mechanism of two-dimensional variations. In addition, we introduce the object number index and the object size index and study their behavior depending on image transformations. In addition, we keep the suppositions that image complexity increases with incrementing the number and the contrast of image details and decreases when the sizes of image details growth.

In the following, we compare known definitions of variations for continuous two-dimensional functions and interpret their reduction for discrete functions. The preference for Kronrod variation is demonstrated. Using two-dimensional variations, some image analysis and processing tasks are explored. Among them are the following: image smoothing with contour preservation, noise filtration, object detection, and image reconstruction. We introduce a new variation-based approach for forming and analysis of both global and local image features and investigate some new qualitative estimates. For the image reconstruction task, we consider the application of metrical and topological image characteristics based on two-dimensional variations. The study accomplishes on the platform of a two-scale image model. Theoretical conclusions are confirmed experimentally.

In Section 3.2, the two-scale image model is stated, which is the base for consecutive image analysis and processing. In Section 3.3, the two-dimensional variations are studied as the apparatus for discrete image analysis. Section 3.4 is devoted to the questions of image complexity estimation. In Section 3.5, the image smoothing and decomposition algorithms are analyzed. Section 3.6 contains the consideration of constructing the topological image characteristics on the base of two-dimensional variations. Section 3.7 is oriented to the application of obtained results to the tasks of image restoration and distortion diagnostics.

3.2 Local Discrete Image Model

An image is a projection of some observed spatial scene to the visual plane. We consider an image as a composition of several connected *areas* that correspond to distinctive details of the scene and densely cover the image space. The spatial area (or *domain*) of the *discrete image* is defined as maximal in size, separately perceived connected set of its elements, having similar, perhaps smoothly varying brightness values and colors. The areas may contact in any manner, including surrounding one another. The brightness (color) values of neighboring domains must noticeably differ. Non-contacting areas may have arbitrary and independent values, including equal ones. The boundaries between neighboring domains are *contours*.

We consider an image as an object, suitable for describing its general properties, features, and characteristics. The formulation of such description composes the *image model* [24], which should accept the lengthy and small domains and contour edges and define the statistical relations of image pixels in distances specified by the task properties. This model should implement the possibilities for developing effective image processing and analysis algorithms.

One of the most important model features is the concept of *analysis region* V_d that is the spatially limited set of image elements, used in the analysis procedure. Usually, only some subset of V_d is interesting that corresponds to an object, containing the *processing element* x_{mn}. This subset is the *belonging area* of the element x_{mn}. The selection of belonging area depends on the task, but we consider it as the intersection of the analysis region and the domain containing the element x_{mn}.

We have no initial splitting of an image to a set of domains. Therefore, the set of belonging area elements should be selected with some chosen method. We use for this goal the statistical methods, based on the distribution of the pixel values in the analysis region, and on the processing element x_{mn}. The methods that for every processed element analyze its own local region, which, as a rule, centered on this element, are named *local methods*.

3.2.1 *Image concepts and models*

3.2.1.1 *Two-scale image concept*

As many researches demonstrate, there is no success in creating a common model, uniformly describing all necessary features of real images. Therefore, we chose the combination of two different models: the *vicinity model* and the *fragment model*, describing image properties in small-sized and large-sized scopes. This approach well corresponds with known data, concerning the structure of receptive fields in visual parts of the cerebral cortex, where the different-scope image analysis is also presented [33]. The vicinity and the fragment models together compose the *two-scale* (*local*) *image model*.

The vicinity model (the scale of elements) considers the connected set of elements, located in the distances of several digitization

steps. The vicinity V_{mn} of the element x_{mn} is the group of R nearest connected elements $x_{mn}^r \in V_{mn}$, $r = 1, \ldots, R$, hitting the same set with x_{mn}.

The fragment model (the scale of objects) describes the properties of an extensive set of elements and may cover several independent image areas. Thus, the aggregate of fragment elements may be a composition of several groups of image elements with different statistical properties.

3.2.1.2 *Multicomponent image concept*

It is convenient to consider an image $x_{mn} \in \mathbf{X}$ as a two-dimensional function, corresponding to certain conditions of smoothness [47] and to regard it as a sum of the following independent components:

$$x_{mn} = s_{mn} + u_{mn} + v_{mn} + \cdots . \tag{3.1}$$

The first term s_{mn} is a piecewise-smooth component which defines the general brightness level of the belonging area of x_{mn} element. The other terms contain information about the texture, small details, etc. Combining them into a unique texture-detail component $t_{mn} = u_{mn} + v_{mn} + \cdots$, without considerable limitation, one can write

$$x_{mn} = s_{mn} + t_{mn}. \tag{3.2}$$

During the acquisition and digitization, an image is distorted by different disturbances like film grain, registration noise, and analog–digital converter violations. Therefore, noise ξ_{mn} is added to (3.2):

$$x_{mn} = s_{mn} + t_{mn} + \xi_{mn}. \tag{3.3}$$

Here all the components are supposed to be independent and additive, and the components t_{mn} and ξ_{mn} are normally distributed and unbiased [15].

3.2.2 *Vicinity model (scale of elements)*

The *vicinity model* describes the statistical relations between the nearest image elements at the distances of several sampling steps.

Let V_{mn} be a vicinity of some shape, rounding the pixel x_{mn}, and containing R elements $x^r_{mn} \in V_{mn}$; $r = 1, \ldots, R$. Denote ρ^r as the distance between the elements x_{mn} and x^r_{mn}. Trace a plane, minimizing mean square error with the elements of V_{mn}; it forms a two-sided angle θ_{mn} with the horizontal plane. Let us assign to each image point (m, n) some vector $\mathbf{g}_{mn}$ of amplitude $g_{mn} = tg\theta_{mn}$ and rotating angle ν_{mn}. Denote the projection of $\mathbf{g}_{mn}$ to the vector from x_{mn} to x^r_{mn} as g^r_{mn}. The traced plane in the point r differs from x^r_{mn} by the random value ξ^r_{mn}. This allows associating the vicinity elements $x^r_{mn} \in V_{mn}$ with the formula:

$$x^r_{mn} = \mu_{mn} + \rho^r g^r_{mn} + \xi^r_{mn}, \tag{3.4}$$

where μ_{mn} is the value of the traced plane in the point (m, n). Tracing a plane using the least square method is not difficult [21]. Let α and β be tangents of plane slopes in vertical (i) and horizontal (j) directions, and (i, j) be the coordinates of a pixel in the vicinity $(-l \le i, j \le l)$. Basing on the equation of a plane $z_{ij} = \mu + \alpha i + \beta j$, formula (3.4) looks as

$$x_{ij} = \mu_{mn} + \alpha i + \beta j + \gamma_{ij}.$$

Having α and β, the amplitude g and angle v of the vector $\mathbf{g}$ are found as

$$g = (\alpha^2 + \beta^2)^{1/2}, \quad v = \mathrm{arc}tg(\beta/\alpha).$$

The amplitude g inside the lengthy areas will be small, but in the contour parts, it will be large. Let us split the ensemble of image pixels into two sets: contour pixels and background (inner) ones. The concept of contour mask $\mathbf{E} = [e_{mn}]$ is the following: $e_{mn} = 0$ for background and $e_{mn} = 1$ for contour pixels. As demonstrated in [15], the distributions of g_c for contour and g_b for background pixels are close to normal ones, but their variances (σ^2) rather differ: $\sigma_c/\sigma_b \approx 3 \div 8$. Therefore, as it is formulated in [15], it is possible to assume background and contour image constituents as the components, established in nonintersecting sets and jointly composing the entire image.

Denote the projection of $\mathbf{g}_{mn}$ to the vector from a central point (m, n) to the point r for contour and background vicinities as φ^r_{mn}

and ψ^r_{mn}. Using the contour mask e_{mn} and substituting g^r_{mn} and ξ^r_{mn} as sums $g^r_{mn} = e^r_{mn}\varphi^r_{mn} + (1 - e^r_{mn})\psi^r_{mn}$ and $\xi^r_{mn} = e^r_{mn}\chi^r_{mn} + (1 - e^r_{mn})\eta^r_{mn}$, the vicinity model formula, describing relations of its elements, will be [15]

$$x^r_{mn} = \mu_{mn} + e^r_{mn}(\varphi^r_{mn}\rho^r + \chi^r_{mn}) + (1 - e^r_{mn})(\psi^r_{mn}\rho^r + \eta^r_{mn}), \quad (3.5)$$

where χ^r_{mn} is the stochastic stimulation in the point r for contour vicinity and η^r_{mn} for background one. Note that the contour mask e_{mn}, in reality, defines the belonging of the central vicinity element x_{mn} to the contour or background pixel set. The random values φ_{mn}, ψ_{mn}, χ_{mn}, and η_{mn} in (3.5) are considered as uncorrelated and unbiased. Furthermore, the values χ_{mn} and η_{mn} may be treated as normally distributed: $N(0, \sigma^2_\chi)$ and $N(0, \sigma^2_\eta)$.

3.2.3 *Fragment model (scale of objects)*

In many tasks, we need to take into consideration the image region in 5–20 times larger than the vicinity size. This needs to develop an appropriate model of the *image fragment*. Let us consider an image as an aggregate of non-intersecting areas, densely covering the image space, and separated by contour borders. According to [42], the general portion of an image is covered by lengthy domains containing from 10^2 up to 10^4 and more pixels. The comparison of necessary fragment sizes and real domain sizes demonstrates that analyzed fragments may entirely fall into one large image domain, may occupy several parts of neighboring domains, or may even completely cover some small domains.

Obviously, the elements of one and the same area are statistically dependable in small distances (inside vicinity), as also in large ones, at least inside one fragment. And vice versa, the brightness and element values of different regions are independent even in small distances.

Consider a fragment W_{mn}, covering R image areas: $U^1, \ldots, U^R$. Denote an element from area U^r with internal coordinates (i, j) inside the fragment W_{mn} as x^r_{ij}. Let us base on the model (3.3), representing an image as the sum of smooth s_{ij}, texture t_{ij}, and noisy ξ_{ij} components. Represent the smoothed components of those parts of R domains, which are inside the fragment, with a polynomial of power

ω (or less). Then the component s_{ij} inside the fragment W_{mn} can be described by the formula:

$$s_{ij}^r = \sum_{U=1}^{R} \delta_{U^r} \sum_{p=0}^{\omega} \sum_{q=0}^{p} a_{pq}^r i^{p-q} j^q, \tag{3.6}$$

where (i, j) is the point of the fragment W_{mn} and δ is the indicator set function: $\delta_{U^r} = 1$, if $(i, j) \in U^r$, and $\delta_{U^r} = 0$ in the other cases. Introducing the texture t_{mn} and noisy ξ_{mn} components into (3.6), we obtain the following expression for the image pixel values:

$$x_{ij}^r = \sum_{U=1}^{R} \delta_{U^r} \left(\sum_{p=0}^{\omega} \sum_{q=0}^{p} a_{pq}^r i^{p-q} j^q + t_{ij}^r + \xi_{ij} \right). \tag{3.7}$$

This is the general formula of the image fragment model, which describes the values of the contained pixel. Note, that for pixels in contour edges, we should use the vicinity model (3.5).

In most real images, the average local brightness of lengthy areas changes enough slowly, so inside typical analysis windows, they can be considered as constant ones; the experiments [15] confirm this. Hence, model (3.7) may be simplified down to the piecewise-constant model with $\omega = 0$. Then $s_{ij}^r(W_{mn}) = s_{mn}^r$, and (3.7) is transformed to

$$x_{ij}^r = \sum_{U=1}^{R} \delta_{U^r} (s_{mn}^r + t_{ij}^r + \xi_{ij}). \tag{3.8}$$

This formula is the piecewise-constant model of image fragment, describing the parts of image areas, covered by the fragment W_{mn}. Note that when the fragment moves, the covered parts change, therefore, generally speaking, $s_{mn}^r \neq s_{m+\Delta m, n+\Delta n}^r$.

As a rule, the values t and ξ are considered as independent, unbiased, and normally distributed: $N(0, \sigma_t^2)$ and $N(0, \sigma_\xi^2)$. Hence, the sum $\tau = t + \xi$ will also be normally distributed: $N(0, \sigma_\tau^2)$, where $\sigma_\tau^2 = \sigma_t^2 + \sigma_\xi^2$. Therefore, formula (3.8) can be simplified to

$$x_{ij}^r = \sum_{U=1}^{R} \delta_{U^r} (s_{mn}^r + \tau_{ij}^r). \tag{3.9}$$

This model is simple, close to the properties of most real images, and is convenient for designing various filtration algorithms, including rank ones. It leads us to use the distribution of pixel values in the fragment as general statistical data for the analysis.

If a fragment covers R parts of different regions, the distribution of its pixel values may have up to R local maxima (modes). The locations of these modes are defined by the average values $s^1, s^2, \ldots, s^R$ of the parts of $U^1, U^2, \ldots, U^R$ regions, covered by the fragment. Portions of the fragment areas, occupied by each region, define relative powers of these modes.

3.2.4 *Color (multispectral) image model*

Color or multispectral image $\mathbf{X}$ is represented as the two-dimensional $M\%N$ array of vectors $\mathbf{x}_{mn} = \{x^1_{mn}, \ldots, x^Z_{mn}\}$. Here, Z is the number of image *channels*, also named as *components* or *zones*. In the case of RGB color image, $Z = 3$. Often, all the images with $Z \geq 2$ are generalized as *color* ones. Usually, the number of channels is clear from the context.

We suppose that every color image channel meets the case of usual single-channel gray-scale image. But for all that, their statistical and informational properties may rather differ. Thereafter, the results obtained during separate channel analysis (for example, detection of contours or regions) may also differ.

By analogy with (3.3), color image $\mathbf{X}$ can be represented as follows:

$$\mathbf{x}_{mn} = \mathbf{s}_{mn} + \mathbf{t}_{mn} + \boldsymbol{\xi}_{mn}, \tag{3.10}$$

where $\mathbf{x}_{mn}$, $\mathbf{s}_{mn}$, $\mathbf{t}_{mn}$, and $\boldsymbol{\xi}_{mn}$ are Z-dimensional vectors. Other image model expressions also can be formulated in vector form. So, similar to (3.9), for the model of color image fragment, we can write

$$\mathbf{x}^r_{ij} = \sum_{U=1}^{R} \delta_{U^r}(\mathbf{s}^r_{mn} + \boldsymbol{\tau}^r_{ij}). \tag{3.11}$$

As before, $\mathbf{x}^r_{ij}$ is the vector value for the element (i, j) of the fragment, centered at (m, n), $\mathbf{s}^r_{mn}$ is the average value for fragment area r, containing the element $\mathbf{x}^r_{ij}$, and $\boldsymbol{\tau}^r_{ij}$ is the vector, defining the disturbance in (i, j).

In many cases, color image transformations can be accomplished independently for every image channel. Nevertheless, in some cases, joint analysis and transformations are necessary.

3.3 Two-Dimensional Variations

Variation is a simple but informative metric characteristic of variability and complexity of 1D-function in an interval $[a, b]$. Considering that function $f(x)$ reflects some real physical value (e.g. energy or density), it is natural to assume that $f(x)$ is bounded on observed interval and has a finite number of first-order discontinuity points. Then, $f(x)$ is a function of the limited total variation, which, by definition, is the following value:

$$V_a^b(f(x)) = \sup_D \sum_{k=0}^{K-1} \|f(x_{k+1}) - f(x_k)\|, \tag{3.12}$$

i.e. the exact upper boundary of partitioning D on the interval $[a, b]$.

3.3.1 *2D variations as the image analysis instrument*

In the transition to the two-dimensional (and all the more, to n-dimensional) case, the difficulties appear with the definition of variation [1, 5]. There were proposed many definitions of multivariable variation (total variation): Arzelá *et al.* [35, 45, 59]. All of them are formulated as a generalization of 1D-variation and give only one value for the analyzed function, which is based, somehow or other, on the modulo of function gradient at a point. Therefore, in spite of the differences in the definitions, their behaviors and values are similar.

Considering an image as a 2D function, Rudin *et al.* [49] explore the possibilities of using the total variation for the tasks of image smoothing and noise removal. These activities arouse significant resonance and generate the continuations, for example [20,25]. Meyer [36] proposed to consider the ROF approach [49] as an algorithm for separating a texture and a *sketch*. Note that such interpretation

fully matches with the above-considered two-scale image model and also with the image decomposition, discussed in Section 3.5.2, which allows splitting an image to piecewise-smooth and texture-noise components.

Kronrod [29] proposed an alternative definition of 2D-variations, based on the concept of the level set of a function; Vitushkin extended this approach for n-dimensional functions [59]. They demonstrate that a function of n variables should be characterized as not one but n independent functionals. The multivariate function variations were studied in [57–59]. An application of Kronrod variations for image processing tasks was studied in [16, 23, 39, 40], where the advantage of this approach was demonstrated.

3.3.2 *Discrete analogs of 2D variations*

For continuous functions, the multidimensional variations are formulated by means of least upper bounds on the set of possible partitioning of the function domain into elementary parallelepipeds by secant hyperplanes. If the function is discrete, its minimum partitioning is bounded below by the accuracy of representation. So, the formulas for discrete function variations are the analog and approximation of the variation formulas for continuous functions. In some cases, this may lead to certain inaccuracies. Furthermore, not all of the operations can be applied to a discrete function without losing the information, e.g. operations of rotation of the function definition area or compression either in the domain area or in the space of values.

Let us consider the adaptation of well-known continuous variations [23, 35, 59] to 2D discrete function $\mathbf{F} = \{f(i,j)\}$ $(i = 1, \ldots, I;$ $j = 1, \ldots, J)$, which is defined on rectangular domain D with corresponding partitioning.

Arzelá variation [35, 59] is, in fact, the sum of gradient modules along coordinate axes and is represented by the formula

$$A(\mathbf{F}, D) = \sum_{i=1}^{I-1} \sum_{j=1}^{J-1} (|f(i+1,j) - f(i,j)| + |f(i,j+1) - f(i,j)|). \tag{3.13}$$

Most of the theoretical and practical elaborations use this variability assessment of the 2D functions.

Vitali variation [35, 59] is formally defined as follows:

$$V(\mathbf{F}, D) = \sum_{i=1}^{I-1} \sum_{j=1}^{J-1} |f(i+1, j+1) + f(i, j) - f(i, j+1) - f(i+1, j)|. \tag{3.14}$$

As a substantial drawback of this variation, we should note that in the area, where the value of function $f(i, j)$ changes only in the direction of one coordinate axis (i or j), the operator (3.14) will be equal to zero.

Pierpont variation [35, 45] assumes the uniform partitioning of domain D to equal segments, i.e. uniform discretization of the definition domain. The oscillation value $\omega(\mathbf{F}, \delta(i, j)) = \{\max f(x, y) - \min f(x, y)\}$, $f(x, y) \in \delta(i, j)$ is measured in every parallelepiped $\delta(i, j)$, and the variation is calculated as

$$P(\mathbf{F}, D) = \sum_{i=1}^{I-1} \sum_{j=1}^{J-1} \omega(\mathbf{F}, \delta(i, j)). \tag{3.15}$$

In the discrete case, the oscillation $\omega(\mathbf{F}, \delta(i, j))$ can be determined as $\omega(\mathbf{F}, \delta(i, j)) = \max\{|f(i+1, j) - f(i, j)|, |f(i, j+1) - f(i, j)|\}$.

Tonelli variation [35, 59] is the sum of 1D variations (3.12) for all the values of each coordinate (i or j), under the fixed value of the second one. Let $\psi_1(x_i)$ be the 1D variation of function $f(i, j)$ in j with $i = \text{const}$, and $\psi_2(j)$ be the 1D variation of function $f(i, j)$ in i with $j = \text{const}$. Then,

$$T(\mathbf{F}, D) = \sum_{i=1}^{I} \psi_1(i) + \sum_{j=1}^{J} \psi_2(j). \tag{3.16}$$

Formulas (3.13) and (3.16) produce equal values, which means that *Arzelá* and *Tonelli* variations coincide for discrete functions.

Frechet variation [35] in the formulation for the 2D discrete function coincides with *Hardy* variation [35, 59] and is calculated as

$$\begin{aligned}\Delta_1(f;(i,j)) &= f(i+1,j) - f(i,j),\\ \Delta_2(f;(i,j)) &= \Delta_2(\Delta_1(f;(i,j)))\\ &= (f(i+1,j+1) - f(i,j+1)) - (f(i+1,j) - f(i,j)),\\ F(\mathbf{F},D) = H(\mathbf{F},D) &= \sum_{i=1}^{I-1}\sum_{j=1}^{J-1} |\Delta_2(\Delta_1(f;(i,j)))|. \end{aligned} \tag{3.17}$$

Note that formulas (3.14) and (3.17) coincide, i.e. the values of *Vitali*, *Frechet*, and *Hardy* variations for discrete functions are equal.

3.3.3 *Two-dimensional Kronrod variations*

In any event, the variations (3.13)–(3.17) are based on the absolute value of the function gradient at a point. Therefore, their values and behavior are similar. All of them come to the following principle: some unique functional is defined, which finiteness guarantees that 2D function possesses some properties, similar to ones of 1D function [59]. However, the list of these properties is much poorer than the collection of 1D function properties. Moreover, depending on the selection of a coordinate system, some ambiguity of multivariate variations appears.

The generalizations of conclusions and theorems, successfully formulated for continuous functions on the basis of the above-defined variations, have led to the following deduction: a function of n variables should be characterized not by one but by n functionals that are independent in a certain sense. Kronrod [29] substantiated this thesis by studying two-variable functions. He proposed to use two functionals that are based on the concept of level sets of a function and not use its gradient characteristics. These functionals are defined as follows:

$$w_1(f) = \int_{-\infty}^{\infty} v_0(e_t)dt, \quad w_2(f) = \int_{-\infty}^{\infty} v_1(e_t)dt. \tag{3.18}$$

Here, set e_t is the t-level of function $f(x, y)$, i.e. the set of the points (x, y), where $f(x, y) = t$, $v_0(e_t)$ is the number of components of the set e_t, and $v_1(e_t)$ is the length of e_t set (according to Hausdorff). The $w_1(f)$ variation is not metric but the topological characteristic of the function f; it is based on the concept of connectivity and remains unchanged under the homeomorphism. In the case of finite $v_0(e_t)$, the value $v_1(e_t)$ may be interpreted as the total length of e_t components. The estimate $w_2(f)$ is analogous to the above-discussed (3.13)–(3.17) total variations [35, 45]. Further, Vitushkin [59] found a felicitous definition for the variation of a set, and this approach, which also is based on the consideration of the level set, was expanded to multivariable functions.

Kronrod variations (3.18), therefore, stand apart in a row of variations of multidimensional functions because it does not use any gradient characteristics of the function and, as a result, gives the values of several (rather than one) functionals.

Concerning the variation value $w_2(f)$, for continuously differentiable function $f(x, y)$, the following equality is true [38]:

$$w_2(f) = \iint_D |\mathrm{grad}(f(x, y))| dxdy,$$

i.e. $w_2(f)$ is the integral of modulo of the function gradient $\mathrm{grad}(f(x, y))$. In a case of discrete function $f(i, j)$, 2D variations (3.18) are expressed as

$$w_1(f) = \sum_{t=0}^{T-1} v_0(e_t)/T \quad \text{and} \quad w_2(f) = \sum_{t=0}^{T-1} v_1(e_t)/T, \tag{3.19}$$

where T is the total number of possible values of function $f(i, j)$ (for an image, it is the number of brightness levels). The calculation of w_1 and w_2 values on bounded domain D assumes that domain D itself is also taken into account as a separate component. This leads to the following contradiction. As obvious, for a constant value function $f(i, j) = \mathrm{const}$, the $w_1 = w_2 = 0$ condition must be true, but, according to (3.19), $w_1(f) = 1$ and $w_2(f) = P(D)$, i.e. equals to the perimeter of domain D. To eliminate this contradiction, formulas

(3.19) should be modified as follows:

$$w_1(f) = \left(\sum_{t=0}^{T-1} v_0(e_t)/T\right) - 1 \quad \text{and} \quad w_2(f) = \left(\sum_{t=0}^{T-1} v_1(e_t)/T\right) - P(D). \tag{3.20}$$

3.3.3.1 *Object quantity index*

The w_1 value, when normalized to the range T, is the characteristic of the quantity and the amplitude of image objects that form separate components in the level sets. It is also possible to demonstrate that the second variation value $w_2(f)$ in (3.20) coincides with the discrete *Tonelli* variation value (3.16).

The values w_1 and w_2 are the estimates of the amplitude and the morphological image characteristics. At that, the first variation, w_1, represents the quantity and the contrast of image details, and the second variation, w_2, represents the sum of their perimeters. Therefore, it is convenient to call the w_1 value as the *object quantity index* of an image.

3.3.3.2 *Object size index*

Along with the values w_1 and w_2, their ratio $q_w(f) = w_2(f)/w_1(f)$ is important because it displays the average perimeter of image details. In a discrete image, the least possible object is one pixel. So, the minimal component in a level set e_t has the linear size of one discretization step, whose perimeter equals 4. This is the minimal possible value for the perimeter/area ratio of the object. Hence, it is convenient to introduce the following characteristic:

$$d(f) = w_2(f)/4w_1(f). \tag{3.21}$$

The $d(f)$ value, by analogy with the object quantity index w_1 in (3.20), will be naturally to call as the *object size index* of an image.

The properties of estimates $w_1(f)$, $w_2(f)$, and $d(f)$ were considered in [16] in the context of image complexity assessment. In particular, it was noted, that because small details and noise pulses are displayed in the level sets as minimal size components, the value d should reduce when the number and the amplitudes of noise pulses and small details increase. It was also demonstrated that $d(f)$

characteristic reflects only the average size of image objects and does not depend on the object quantity and contrast; this part of the information is carried by the value $w_1(f)$.

3.4 Image Complexity

3.4.1 *On the complexity of signals and functions*

The complexity notion is one of the important concepts for signals, functions, and flow traffic; in the following, we consider its applicability to the image data.

A *1D signal*, which is a real function of time, is characterized by its energy, duration Δt, and spectrum width Δf [57]. Formally, Δt and Δf cannot be finite simultaneously. For real signals, which have the onset and the end in time, the spectrum width is the frequency band in which almost all energy (perhaps, 99%) is concentrated. In theoretical studies, Δf is often assumed as the position of the first zero of the spectrum. It is known from signal theory that, irrelative to the duration definition and to the signal spectrum width, the minimal value of the product $\Delta f \Delta t$ (which is called as "*base*" for simple signals) approximately equals unity [27]. On the contrary, the signals with $\Delta f \Delta t \pi 1$ are called complex ones [57]. However, it is difficult to apply this definition to images. First, an image is a 2D function, and second, the image points correspond to the same time instant or interval.

In accordance with [58], the complexity of an *analytical function* of many variables is characterized by the number of variables n. For s-fold differentiable functions, it is defined by the ratio n/s. This approach is hardly applicable to an image as a brightness function of spatial coordinates, since, in the case of images, we have only $n = 2$.

The *information-theoretical* approach to the assessment of the discrete flow traffic complexity is based on the calculation of entropy value: $H = -\Sigma_b p(z) \log(p(z))$. Here, $p(z)$ is the probability of the image element with the value z, and $b \in [b_{\min}, b_{\max}]$ is the range of z. In fact, this variant assumes the independence of image point values, while the image is a signal with strong spatial relations.

So, the approaches used in evaluation of the complexity of the flow of independent traffic, e.g. entropy, are inconsistent.

For evaluation of the complexity of a discrete signal, the coefficient of *reduction of the data volume* is often used; here *data volume* means the number of bits required for signal restoration. This approach, which is ideologically close to the Kolmogorov complexity concept [28], is associated with the necessity to select one or another compression algorithm and therefore is similar to the above entropy evaluation.

Due to one or another reason, every of the above complexity evaluation methods is less or even not applicable for an image as a two-dimensional function of spatial coordinates, which characterizes some physical parameters of a scene. In particular, substantial drawbacks of the above estimation variants arise from the fact that these characteristics are not metric and are independent of the signal amplitude.

3.4.1.1 *Using the complexity estimation in restoration task*

The numerical estimate of a signal is important in solving various problems, in particular, in signal restoration problem [52] (see Section 3.7). Let a signal z be distorted by operator A and by noise ξ, and the signal $u = A(z) + \xi$ is the result. Assuming that operator A is known with a certain accuracy, the restoration operator $\hat{A}(u) \Rightarrow \{z_\alpha\}$ is constructed, which maps u onto a set of admissible results $z_\alpha \in Z$. The following functional is calculated:

$$\varphi_\alpha(u, \lambda) = \|Az_\alpha - u\| + \lambda\Omega(z_\alpha), \tag{3.22}$$

where $\Omega(z_\alpha) > 0$ is the stabilizing functional and λ is the regularizing parameter. The result z^*, optimal in Z, is calculated by minimizing the functional $\varphi_\alpha(u, \lambda)$ with selected λ:

$$\varphi_{z*}(u, \lambda) = \inf_{z^\alpha \in Z} \varphi_\alpha(u, \lambda). \tag{3.23}$$

The restoration operator $\hat{A}$ is constructed based on the assumptions of the properties of distorting operator A and depends on specific conditions. For example, if the restoration problem is linear [52],

it can be inverse operator $\hat{A} = A^{-1}$; in some others, such as in the noise filtration problem based on the total variation [49], it can be the unit operator $\hat{A} = E$, and so forth.

It is important, that in contrast to the restoring operator $\hat{A}$, the stabilizing functional $\Omega(z)$ in (3.22) does not depend on the distorting conversion. Hence, it can be selected as a characteristic of restored signal z, which can be specified from the signal model. For 1D function, it is usually proposed to use estimates like a norm or variation [52]. It is useful to note that the functional $\Omega(z)$ can be also interpreted as some signal complexity estimate.

3.4.2 *2D variation as an image complexity assessment*

Let us analyze the object size index d, defined by (3.21). For an image with the dominance of small details, represented in a level set by low-sized components, value d will be small. For an image with gross details, value d will be large. Noise pulses, which are local signal spikes, are treated as image details and are interpreted as separate components of minimal size. Therefore, the value d should be reduced by increasing the number and the amplitude of noise spikes. In the following, this fact is confirmed experimentally.

The behavior of $w_1(f)$ and $d(f)$ values under possible transformations of the function $f(x, y)$ is of interest. For several important cases, the following statements can be formulated [14]:

(i) The w_1 value does not depend on the coordinate system, but d slightly depends on the selection and on the turn of the coordinate system with regard to domain D and according to the sampling accuracy of $f(x, y)$.
(ii) Under the linear amplitude transformations of type $Cf(x, y)$, where C is a constant, w_1 increases proportionally to C and d remains unchanged.
(iii) In linear spatial stretching of the domain D by a factor K, the value w_1 does not change, but d increases proportionally to K.
(iv) Let D be expanded by adding the domain U, where $f(x, y)$ has a constant value: $f(x, y) = \text{const}$, $\{(x, y) \in U \subset D' = D \cup U\}$. The condition for this expansion operation is that it should not add

new first-order discontinuities. Such expansion does not modify w_1 and d.

(v) When the number of image details increases (under preserving statistical features), the w_1 value proportionally increases, but d does not change.

(vi) Let the domain D be rectangular and bounded by the interval $[a, b]$ on the selected axis, say the Y-axis. Let us expand it by adding the domain U, specified in half-interval $(b, 2b-a]$, where $f(x, y)$ has the mirror continuation: $f(x, b + y) = f(x, b - y)$. In this expansion, w_1 increases proportionally to the change of domain area, while d does not change.

Therefore, with the above changes of 2D function $f(x, y)$, at least one of w_1 and d values proves to be invariant, and the second value is a predictably transformed function. As applied to an image, the statement corresponds (i) to image rotation, (ii) to linear contrast change, (iii) to linear geometric transformation, and (iv) to the addition/deletion of image parts with a flat background without details; the statement (v) is close to the statement (iv) and corresponds to concentration/rarefaction of image details; and the statement (vi) is the variant of mirror image prolongation beyond the bounds of domain D, which is used by many algorithms. In addition, the prolongation of the properties of the image portion to the whole image means signal homogeneity. It means the sufficiency of the determination of w_1 and d in the confidential section and their interpretation of the image as a whole.

Let us note the important role of object size index $d(f)$, which is used in relations (ii) and (v). The index $d(f)$ reflects only the average size of image objects and does not depend on the quantity and the contrast of objects itself. This part of the information is carried by the $w_1(f)$ variation.

The conclusion is that the first variation $w_1(f)$ with the characteristic $d(f)$ reflects the spatial and brightness variability of the image and, in the aggregate, can act as *image complexity* indices in accordance with the above-formulated requirements. The behavior of these characteristics confirms Kronrod's hypothesis that for the description of 2D function variability, one variation value is not enough, but two independent functionals are necessary.

3.4.2.1 *On comparison and interpretation of assessments*

Since we have two characteristics of complexity, which, generally speaking, are independent, the question arises of how to compare different images using these data. It is rather evident that, if the characteristic $d(f)$ is constant, the image complexity will grow up with the increase in $w_1(f)$. Also, the complexity will grow up with the decrease in $d(f)$, while $w_1(f)$ is constant. Then, if for the images f_1 and f_2 the relationships $w_1(f_1) < w_1(f_2)$ and/or $d(f_1) > d(f_2)$ are observed, it is natural to assume that the image f_2 is more complex than f_1. The question remains on how to compare the other cases.

Let us consider an image with small w_1 and d; it means that the image contains small-sized details, and, either their quantity is low or they are low-contrast. On the contrary, large w_1 and d indicate that the image contains large-sized details and probably they are high contrast.

In various cases, it is necessary to combine some generalized functional from several independent values. One of the simplest versions is the following linear combination:

$$w(z) = w_1(z) + Cw_2(z) = w_1(z)(1 + Cd(z)), \tag{3.24}$$

which can be applied in (3.22) as a stabilizing functional $\Omega(z)$.

3.4.3 *Experimental studies*

It is of interest to study the alterations of image complexity indices $w_1(f)$ and $d(f)$ from image to image: how they depend on the noise level and how they change under image smoothing. For that, we need to compare the transformed signal with some undistorted (etalon) one. In composing such an etalon image, we base it on two-component image model (3.9). In accordance with it, an image $f(i,j)$ implies a sum of two components: piecewise-smooth $s(i,j)$ and texture-noise $\tau(i,j)$ ones. The last component carries information about image texture, fine details, and noise:

$$f(i,j) = s(i,j) + \tau(i,j) = s(i,j) + t(i,j) + \xi(i,j). \tag{3.25}$$

The property of local algorithms is that the processing result value at every image point (i,j) depends only on the elements in some



Figure 3.1. Test images of 256×256 pixels.

limited analysis area $R(i,j)$. In Section 3.2.3, it was demonstrated that for the majority of real images, we may accept $s(u,v) = s(i,j)$, if $(u,v) \in R(i,j)$. Therefore, within the limits of $R(i,j)$, the value $s(u,v)$ is considered as a constant. It allows composing an etalon image as a piecewise-constant 2D signal, which can be then distorted and transformed.

For illustration, we picked out a set of real and synthesized images with different details and thematic content. Some images are shown in Figure 3.1. There are portraits with fine (a) and coarse (b) details, urban landscape (c), air photo (d), still life (e), computer tomogram (f), text fragment (g), and synthesized signal (h) with details of different size and brightness.

Table 3.1 summarizes the values of 2D variations for images in Figure 3.1. Two-dimensional Arzelá (A), Tonelli (T), Pierpont (P), Vitali (V), Frechet (F), Hardy (H), and Kronrod (K-w_1 and K-w_2) variations were measured according to formulas (3.13)–(3.17) and (3.19). The object size indices $d(f)$ according to (3.21) and entropy values (E) are also given.

The values of the same variation can vary for different images 4 times and more, while the entropy differs very weakly. This confirms that the entropy cannot be an appropriate image complexity assessment. Comparison of the values in Tables 3.1 and 3.2 once again confirms the coincidence of Arzelá/Tonelli (A/T) and second Kronrod ($K - w_2$) variations.

Table 3.1. Values of 2D variations, object size index (d), and entropy (E) for the images in Figure 3.1.

Image	A/T	P	$V/F/H$	K-w_1	K-w_2	d	E
a	5242.72	3715.70	2859.43	469.85	5260.25	2.80	7.59
b	2583.56	1825.14	1038.43	129.23	2589.58	5.01	7.60
c	7928.75	6072.46	3497.91	618.79	7947.70	3.21	7.59
d	4804.46	3399.38	2323.97	380.03	4804.38	3.16	7.16
e	3777.20	2844.99	1497.63	178.63	3777.15	5.29	7.29
f	5107.64	3642.48	2499.84	419.62	5118.65	3.05	6.76
g	6462.14	4989.61	2165.84	231.24	6462.04	6.99	5.61
h	811.67	778.54	132.52	33.13	811.67	6.12	0.58

Table 3.2. Values of 2D variations, object size index (d), and entropy (E) for the image in Figure 3.1(h) depending on the noise level.

Noise (scales)	A/T	P	$V/F/H$	K-w_1	K-w_2	d	E
$\sigma = 0$	811.67	778.54	132.52	33.13	811.67	6.12	0.58
$\sigma = 1$	1376.84	1163.84	544.80	133.52	1378.40	2.58	2.68
$\sigma = 2$	1938.47	1538.89	944.20	233.46	1943.44	2.08	3.63
$\sigma = 3$	2505.54	1918.21	1347.78	333.57	2511.99	1.88	4.21
$\sigma = 5$	3633.55	2679.20	2161.07	534.40	3644.56	1.70	4.93
$\sigma = 10$	6478.16	4568.57	4171.93	1035.24	6478.06	1.56	5.86

The variation and entropy values for a synthesized image (Figure 3.1(h)) and similar ones essentially differ from corresponding values for real images. The reason is that such an artificial piecewise image does not contain both noise and very thin details. If we add even minor noise, which denotes as ξ component in (3.3), the estimates grow up to be similar for real values. The noise fluency is considered thoroughly in the following.

3.4.3.1 *The noise fluency to the image complexity assessment*

The synthesized image (Figure 3.1(h)) with details of different brightness and size was used for estimating the fluency of the noise to the image complexity assessment. It was distorted by Gaussian noise $N(0, \sigma^2)$ with the *rms* deviation $\sigma = 1, 2, 3, 5$ and 10 grayscales. Table 3.2 demonstrates the changes in the values of different 2D variations, object size index (d), and entropy (E) depending on the

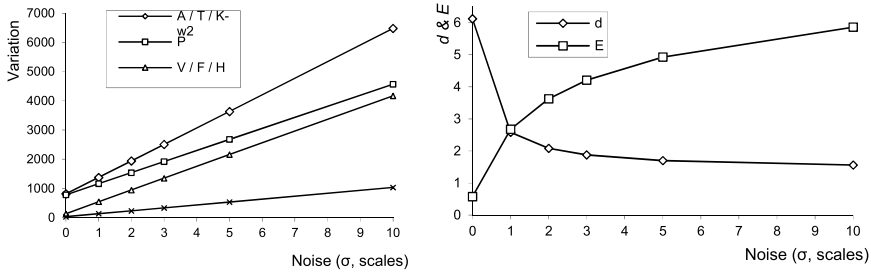


Figure 3.2. Dependences of the variation (*Variation*), object size index (d), and entropy (E) on the *rms* deviation of the added Gaussian noise (σ, *gray scales*).

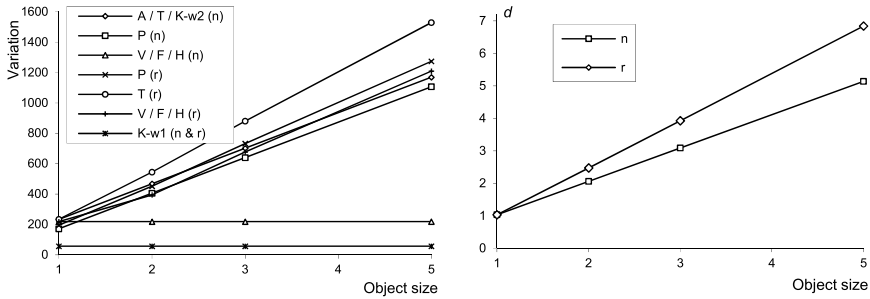


Figure 3.3. Dependences of the variation (*Variation*) and object size index (d) on the *Object size* in the normal (n) and rotated (r) positions.

noise level. The values from Table 3.2 are also displayed graphically in Figure 3.2. It should be noted that even weak noise with $\sigma = 1$ grayscale significantly changes the values of all variations.

Based on the mentioned data and dependences, one can make the following conclusions: (i) all the variation values increase linearly with ascending in the noise level, (ii) the object size index (d) primarily steeply drops, then reduces fluently tending to the limit value of 1, (iii) the entropy (E) increases asymptotically to a limit, equal to the bit number in image pixel (8), and (iv) the Arzelá/Tonelli (A/T) variations and the second Kronrod (K-w_2) variation coincide with a high accuracy, thereby confirms the above inferences.

3.4.3.2 *Dependences of variations on stretching and rotation*

Figure 3.3 demonstrates the dependences of 2D variations on detail size and rotation. For this, several images similar to Figure 3.1(h) were formed. Images contain an identical number of rectangular

random-brightness objects, sized from 1×1 to 5×5 pixels, on the constant background. Variations and object size indices were measured both for images with the normal position of the objects, when their sides are parallel to the rows and columns, and also for images rotated by 45° objects without interpolation of element values.

As one can see in Figure 3.3, the graph $K\text{-}w_1(n\&r)$ is horizontal and coincides with the normal (n) and rotated (r) objects. This confirms that the object number index w_1 is invariant to both changes in the object sizes and in their rotation. The constancy of the $V/F/H(n)$ graph, which corresponds to the objects in their normal orientation, confirms the above conclusion that the local operators used in discrete analogs of Vitali/Frechet/Hardy variations give zero responses on those contour lines of two-dimensional objects, where it has the value drop in the direction of one coordinate axis only. For not vertical or not horizontal contour lines, this property is not accomplished (graph $V/F/H(r)$). The graphs of other variations are located closely, pointing that the corresponding variation values differ weakly.

As it is obvious from the right graphs in Figure 3.3, the object size index (d) varies proportionally to the object size, thus illustrating the statement (iii) in Section 3.4.2 that the value of this characteristic depends linearly on the domain stretch factor. The fact that the graph, corresponding to rotated objects, has a slightly higher steepness, thus contradicting the statement (i), is explained by contour length changing. Really, the contour length of the rotated discrete object, which is measured in metric L_1, is longer than the contour length of the object with vertical and horizontal edges.

The above results demonstrate that obtained assessments w_1 and d reflect important image characteristics like a number, sizes, and visibility of image details and therefore may serve as the assessment of the image complexity. This assessment can be used as the resulting image estimate, for example, in stabilizing functional under solving the signal reconstruction tasks.

3.5 Decomposition of an Image to Information Components

According to the model discussed in Section 3.2, an image is the additive mixture of several components with different information

contents. Splitting an image to a set of constituents, each being less complicated than source signal and containing distinct information, simplifies posterior image analysis and processing. We consider the task of splitting an image into two components according to model (3.2): piecewise-smooth component s_{mn}, containing lengthy areas and borders between them (*smoothed* component), and texture-detail component t_{mn}, containing texture, fine details, and noise (*texture* one). Such splitting we name as *decomposition* [8].

The extraction of the component of interest opens new possibilities in deciding several image processing and analysis tasks. For example, extracting the texture component and removing the information about average brightness simplify the texture segmentation procedure. Varying the image separation parameters, object detection tasks can be realized [10].

Since the component s_{mn} is smooth in inner parts of image regions, its extraction may be interpreted as independent smoothing of every image area and named as *edge-preserving image smoothing.* The component, complying with the above goal, meets the requirements of the task of finding the minimal complexity image under retention of the information about contained objects.

3.5.1 *Edge-preserving smoothing*

Signal smoothing is one of the widespread operations for noise reduction. However, along with removing the noise, most of the smoothing methods blur image contours. Numerous researches dedicated to discovering the method, which along with sufficient smoothing ensures the preservation of contour edges. Among them are median filter [56], adaptive linear filter [31], mask method [41], gradient inverse weighted smoothing [60], smoothing in selected neighborhoods [54], sigma filter [32], and others. The comparison of these methods was discussed in [34]. In all the methods, smoothing is considered only as a noise-removing procedure. From the above-discussed image model, it is clear that image smoothing is appreciably more powerful.

To clarify the difference between the image smoothing over small and lengthy areas, we use the image model, considered in Section 3.2. In the first case, according to (3.5), smoothing gives the value μ of the vicinity, while in the second case, it should specify the s component value in (3.8). We are interested in the second task, which is the estimation of s component under the edge preservation conditions.

We base on the above-mentioned two-scale image model, which is local. Therefore, the smoothed value estimation for the element x_{mn} is totally defined by its surrounding area $V_d(x_{mn})$. Following the approach of separate smoothing of every area, we need to find the method for segregation from the total analysis area some subset of elements, which constitute the *belonging area* for the element x_{mn}.

There is no preliminary splitting of an image to a set of necessary areas; therefore, the selection of elements, related to the belonging area, is realized with statistical methods based on the pixel values distribution in the analysis area $H_{mn}^{V_d}$. This distribution is often called the *histogram of the local fragment* or, simply, the *local histogram*:

$$H_{mn}^{V_d} = \{h_{mn}(k)\}, \tag{3.26}$$

where $h_{mn}(k) = P\{x = k | x \in V(m,n)\}$; $0 \le k < K$; $\sum_{k=0}^{K-1} h_{mn}(k) = 1$.

The advantage of using the local histogram as the principal instrument for analysis is based on a simple and effective algorithm for calculating the histogram and order statistics for sliding fragments in the transition from element n to element $n+1$.

It is demonstrated in the following that finding the necessary belonging area corresponds to the selection of some range of values $[x_l, x_r]$ in the $H_{mn}^{V_d}$ distribution, being used for estimation. This means the introduction of the *locality* also into the value domain. Considering the spatial and value domains together, the methods that use the dataset, local on two different characteristic types, are generally mentioned as *bilateral methods*.

3.5.1.1 *Smoothing on the base of distribution analysis*

According to model (3.2), let us consider image smoothing as the task of finding the piecewise-smooth component s_{mn} and also restrict with the class of methods based on the probability distribution over the fragment (local histogram) $H_{mn}^{V_d}$, $V_d = W_{mn}$. Let W_{mn} include some parts of U^j $(j = 1, \ldots, J)$ image areas. According to the fragment model (3.9), image smoothing task can be reformulated as finding out the average brightness s^1 of the area U^1 of the fragment W_{mn} that contains the central element x_{mn}. The distribution $h_{mn}(k)$ contains not more than J local maxima (modes). Their placements are defined by the set of mean values s^j, and the powers are defined by

their probabilities in $h_{mn}(k)$ distribution, i.e. by the area portion of region j in the fragment W_{mn}. The central fragment region U^1, which contains the element x_{mn}, may correspond to any mode of the histogram $H_{mn}^{V_d}$. Therefore, the retrieving of s_{mn}^1 value, i.e. the average for the region U^1, can be interpreted as finding out the mode in the histogram $H_{mn}^{V_d}$, corresponding to U^1.

This statement is typical for the recognition and classification tasks based on statistical models [18]. One of the acceptable Bayes classifiers is the Parzen estimation window method [43], which uses the following function:

$$f(u) = \frac{1}{Kh} \sum_{k=0}^{K-1} \Psi \left(\frac{x_k - u}{h} \right), \tag{3.27}$$

where $\Psi(x)$ is the kernel analysis function and h is its effective width.

There are many classification algorithms based on this idea, for example, the nearest-neighbor one. As is known [30], the fluency of the shape of function Ψ is powerless, whereas the width parameter h is determinative. This can be used for estimating an average for the belonging area as follows:

$$s_{mn} = \sum_{k=0}^{K-1} x_k F(x_k - x_0) \Bigg/ \sum_{k=0}^{K-1} F(x_k - x_0), \tag{3.28}$$

where $F(x)$ is the weight function and x_0 is the reference value for (m, n).

3.5.1.2 *Sigma filter*

If the central vicinity element x_{mn} is used as x_0 in (3.28), that is, $F_i(x) = 1$ for $-\delta \le x \le \delta$ and $F_i(x) = 0$ for others x, we obtain the well-known *sigma filter Lee* [32]. Its essence is, that from vicinity elements $x_{mn}^k \in V'_{mn}$ only the elements x_{mn}^r $(r = 1, \ldots, R)$ in the interval $(x_{mn} - \delta, x_{mn} + \delta)$ are chosen. Here δ is the width parameter. From x_{mn}^r values, the result of smoothing s_{mn} is found as

$$\begin{aligned} s_{mn} &= \mathrm{A}(V'_{mn}, x_{mn}, \delta) \\ &= \frac{1}{R} \sum_{r=1}^{R} x_{mn}^r, \quad \text{for } x_{mn} - \delta \le x_{mn}^r \le x_{mn} + \delta. \end{aligned} \tag{3.29}$$

In [32], it is recommended to use $\delta = 2\sigma$, where σ^2 is the noise variance.

In [34], the sigma filter Lee is recognized as one of the best edge-preserving smoothing filters for the vicinities of $3 \div 7$ elements in size. Its disadvantage is that x_{mn} value is an insufficient estimation for noisy images. With significant probability, the range $-\delta \le x_{mn} \le \delta$ may not capture the desired distribution peak or, vise versa, may cover several peaks, corresponding to different regions. In these situations, estimate (3.29) will be incorrect.

3.5.1.3 *Bilateral filter*

Bilateral filtering algorithm [53] is, *per se*, the extension of the above-written sigma filter (3.29) by possibilities of alteration of the shapes of weight function $F(z)$ and analysis area $w(d)$. For that proposed in [53] Gauss shapes of $F(z)$ and $w(d)$, the bilateral filter may be written as follows:

$$F(z) = \frac{1}{\sigma_F\sqrt{2\pi}} e^{-\frac{(z-x_0)^2}{2\sigma_F^2}}; \quad w(d_r) = \frac{1}{\sigma_w\sqrt{2\pi}} e^{-\frac{d_r^2}{2\sigma_w^2}},$$

where $d_r^2 = (m_r - m)^2 + (n_r - n)^2$, then:

$$s_{mn} = \sum_{r\in W_{mn}} e^{-\left(\frac{(w^r-x_0)^2}{2\sigma_F^2}+\frac{d_r^2}{2\sigma_w^2}\right)} x_{mn}^r \Bigg/ \sum_{r\in W_{mn}} e^{-\left(\frac{(w^r-x_0)^2}{2\sigma_F^2}+\frac{d_r^2}{2\sigma_w^2}\right)}. \tag{3.30}$$

3.5.1.4 *Rudin–Osher–Fatemi (ROF) smoothing*

The *ROF algorithm* [49] is offered and declared as the method for noise removing, supposing the image f as the sum $f = u + v$, where u is the "pure" image and v is the additive noise. For finding the component u, the following functional is minimized with tuning parameter $\lambda > 0$:

$$Z(u) = \int_D |\nabla u| + \lambda \int_D (f-u)^2 = \int_D |\nabla u| + \lambda \int_D v^2, \tag{3.31}$$

where $\int_D |\nabla u|$ means total variation of u in D. In [36] this algorithm is interpreted as the texture separation algorithm. In [20] u component is named as "sketch", and v is considered as the mixture of

texture and noise. Also supposed that image objects in f belong to u and are limited by finite length contours. It allows considering u as the function of bounded variation.

This treatment of u and v components entirely corresponds to s and $(t + \xi)$ components in (3.3), or, to s and τ components in (3.9). Therefore, ROF algorithm is oriented for splitting an image into piecewise-smooth u and texture-noise v components. This view adjusts with image model (3.9).

3.5.2 *Decomposition algorithm*

The choice of the closest to x_{mn} mode from the distribution H^V_{mn} may not be correct. The reason is that the distribution width σ_τ of τ component in (3.9) for real images may be close to or even exceed the difference $|s^1 - s^j|$ between the average brightness of neighbor areas in the fragment. Hence, it is probable that x_{mn} will be close not to the brightness of the belonging area but to the brightness of some another, that is, wrong area. The analysis demonstrates, that dependingly on the hitting of the (m, n) point to a homogeneous domain or to its border, the variances σ_t and σ_ξ of the texture and noise components in (3.8) may differ 2–3 times. Therefore, the value μ_{mn} in (3.5) will be more correct estimate for s^1_{mn}. This leads us to the conclusion that μ_{mn} value, which can be obtained with the analysis of vicinity V_{mn}, should be used as the preliminary estimation of s^1_{mn} rather than x_{mn} value.

Based on the above, it is possible to formulate the following image smoothing algorithm, preserving the contour edges, as the *decomposition algorithm* that contains two stages of analysis procedure [8–11].

Let the following be specified: the sizes $l \times l$ of the vicinity V_{mn} and $L \times L$ $(l < L)$ of the fragment W_{mn} centered at (m, n), the brightness intervals Δ^V and Δ^W width, and the clipping thresholds $n^V < l^2/2$ and $n^W < L^2/2$. The following algorithm allows finding the value s_{mn} in (3.9). It includes the following operations, executed for every image point (m, n):

(i) The distributions of pixel values over the vicinity H^V_{mn} and over the fragment H^W_{mn}, centered at (m, n), are calculated.
(ii) Based on H^V_{mn} and n^V, the clipping parameters R^V_1 and R^V_2 are found: $R^V_1 = R^V(n^V/l^2)$ and $R^V_2 = R^V(1-n^V/l^2)$; here $R(x)$

is the decision of the equation $\sum_{i=0}^{R(x)} h_{mn}^V(i) = x$. Then the clipped value $\tilde{x}^V$ is evaluated by comparison x_{mn} with R_1^V and R_2^V: $\tilde{x}^V = x_{mn}$, if $R_1^V \le x_{mn} \le R_2^V$; $\tilde{x}^V = R_1^V$, if $x_{mn} < R_1^V$; and $\tilde{x}^V = R_2^V$, if $x_{mn} > R_2^V$.

(iii) The average value $\bar{x}_{mn}$ of the values $x_{mn}^i \in V_{mn}$ $(i = 1, \dots, I)$ in the interval $(\tilde{x}^V - \Delta^V, \tilde{x}^V + \Delta^V)$ is found as

$$\begin{aligned} \bar{x}_{mn} &= \mathrm{A}(V_{mn}, x_{mn}, n^V, \Delta^V) \\ &= \frac{1}{n} \sum_{r=1}^{n} x_{mn}^r, \quad \tilde{x}^V - \Delta^V \le x_{mn}^r \le \tilde{x}^V + \Delta^V. \end{aligned} \tag{3.32}$$

(iv) Similar to (ii), the values $R_1^W = R^W(n^W/L^2)$ and $R_2^W = R^W(1-n^W/L^2)$ are found using the distribution H_{mn}^W and specified n^W. The value $\tilde{x}^W$ is evaluated by comparison $\bar{x}_{mn}$ with R_1^W and R_2^W: $\tilde{x}^W = \bar{x}_{mn}$, if $R_1^W \le \bar{x}_{mn} \le R_2^W$; $\tilde{x}^W = R_1^W$, if $\bar{x}_{mn} < R_1^W$; and $\tilde{x}^W = R_2^W$, if $\bar{x}_{mn} > R_2^W$.

(v) Smoothed value s_{mn} is evaluated over H_{mn}^W as the mean or median of those elements from W_{mn} that are located in the interval $(\tilde{x}^W - \Delta^W, \tilde{x}^W + \Delta^W)$:

$$\begin{aligned} s_{mn} &= \mathrm{A}(W_{mn}, \bar{x}_{mn}, n^W, \Delta^W) \quad \text{or} \\ s_{mn} &= \mathrm{med}(W_{mn}, \bar{x}_{mn}, n^W, \Delta^W). \end{aligned} \tag{3.33}$$

The result, obtained in (3.33), is considered as the smoothed value s_{mn} in (3.9) for the point (m, n). The choice between the mean and the median in (3.33) can be done based on the information that calculation of the mean is simpler, but the median gives a more precise estimate near edges.

3.5.2.1 *About the similarity of local filtering algorithms*

Local average and local median are the most often used filters [11,19]. On the other hand, Parzen estimation (3.27) is the basis for the filters that are local in the values domain. The bilateral filters are, *per se*, the combination of these two approaches. Sigma filter Lee (3.29) uses rectangular weighting functions in the spatial area, as in the value domain. Bilateral filter (3.30) differs from the sigma filter by using Gauss weighting functions. ROF algorithm (3.31) splits an

image to piecewise-smooth u and texture-noise v components and this approach adjusts with the considered image model (3.9).

Image decomposition algorithm (3.33) uses a double filtration kernel: first, over the vicinity V_{mn} of the element x_{mn}, and second, over the surrounding fragment W_{mn}. Using n^V and n^W, parameters allows us to distinguish the image pixel as owned to texture or to piecewise-smooth component [13].

3.5.3 *Smoothing and decomposition experiments*

In the introduction to this chapter, we note that finding the smoothed component with decomposition algorithm is analogous to finding the minimal complexity image under the condition of preservation of the maximum information about the objects. In Section 3.4, we demonstrated that an estimate, based on the two-dimensional variations, is a suitable measure for image complexity. In Section 3.4.3, we display the influence of the noise on the image complexity.

The behavior of image complexity estimate after the image smoothing and decomposition is of interest. The experiments were carried out both with noised test images, having undistorted originals, and with real images.

3.5.3.1 *Experiments with test images*

The piecewise-constant test image, containing numerous objects of different sizes and contrasts (Figure 3.1(h)), was used for the exploration of the dependencies of image complexity on smoothing algorithms. This image was distorted by additive Gauss noise with $\sigma = 5$ grayscales. The following smoothing algorithms were used for noise removing and image restoration: local average, local median, Gauss filter, and image decomposition (3.33).

The results of the experiments are shown in Table 3.3 as the values of object number index (w_1), second Kronrod variation (w_2), and object size index d. Line 1 corresponds to the source clear image and line 2 to the noise-distorted one. Other lines display the results of filtering the noisy image by different smoothing algorithms with equivalent analysis fragment of 15×15 pixels: lines 3 and 4 are local averages and medians, line 5 is a Gauss filter, and line 6 is the smoothed component s_{mn} after the decomposition (3.33).

Table 3.3. Values of MSD, object number index (w_1), second Kronrod variation (w_2), object size index (d), and entropy (E).

Image/Estimate	MSD	w_1	w_2	d	E
Source image	0.00	33.13	811.87	6.13	0.58
Noised, $\sigma = 5$ scales	5.00	533.71	3644.04	1.71	4.93
Local mean	19.30	8.83	463.95	13.13	4.08
Local median	20.26	6.80	253.46	10.51	2.09
Gauss smoothing	18.91	4.40	340.26	14.71	4.10
Decomposition	0.64	36.55	872.95	5.97	1.68

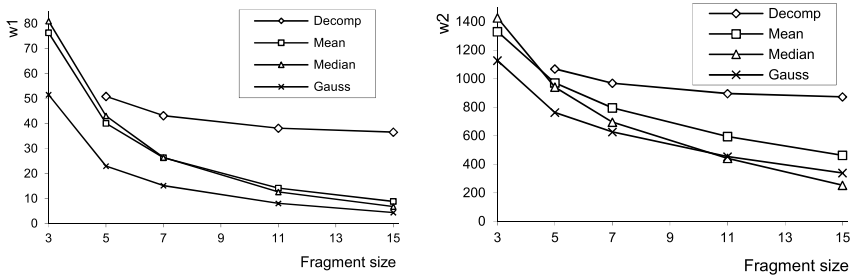


Figure 3.4. The changes of the object number index (w_1) and the second Kronrod variation (w_2) depending on the *Fragment size*.

The changes of object number index w_1, second Kronrod variation w_2, object size index d, and MSD of the filtered image from the non-distorted original depending on smoothing window size are illustrated in Figures 3.4 and 3.5. The fragment size varied from 3×3 to 15×15 pixels. The results are given for local mean (*Mean*), local median (*Median*), Gauss smoothing (*Gauss*), and decomposition (*Decomp*) algorithms.

To compare smoothing methods in the case of strong noise, we use a test image with two extensive regions differenced in 100 grayscales, distorted by Gauss noise of $\sigma = 30$ scales (Figure 3.6(a)). This image was filtered by the above-discussed methods. The estimates for (a) image are the following: $w_1 = 3063$, $w_2 = 17331$, $d = 1.41$, $\sigma = 30$. The results of smoothing with 21×21 pixel window and 4 iterations are shown in Figures 3.6(b)–3.6(d), accompanied by the graphs of the same line. Images (b), (c), and (d) were processed by sigma filter (3.29), bilateral filter (3.30), and decomposition algorithm (3.33)

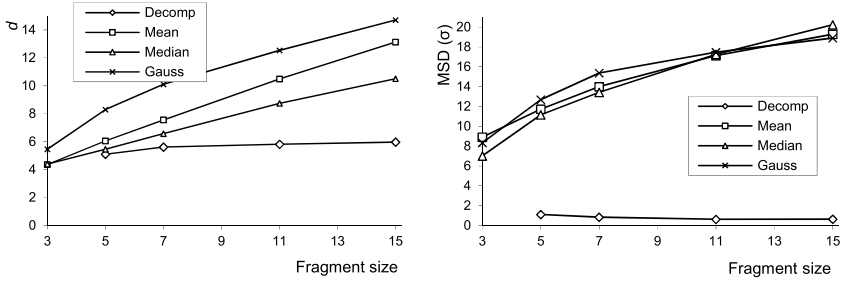


Figure 3.5. The changes of the object size index (d) and the mean square deviation (MSD) depending on the *Fragment size*.

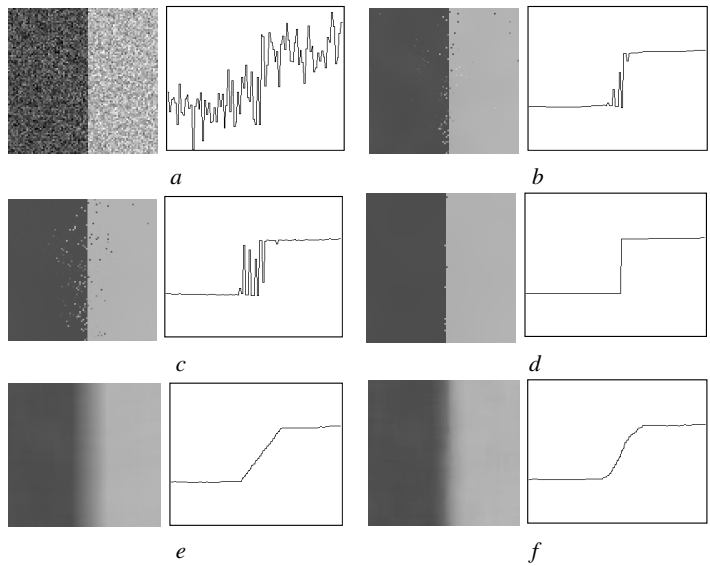


Figure 3.6. Image smoothing by different filters: (a) noisy image; (b) sigma filter; (c) bilateral filter; (d) decomposition filtering; (e) local mean; (f) local median.

correspondingly. For the comparison, images (e) and (f) demonstrate single local mean and local median smoothing with the same window size. For a numerical estimate of the result, the mean square deviation (MSD) of the smoothed image from the original image without noise is used. For constant-regions image, this estimate is acceptable.

The dependences of signal recovery precision (MSD) on the filter type and on the window size, measured for the image in Figure 3.6(a),

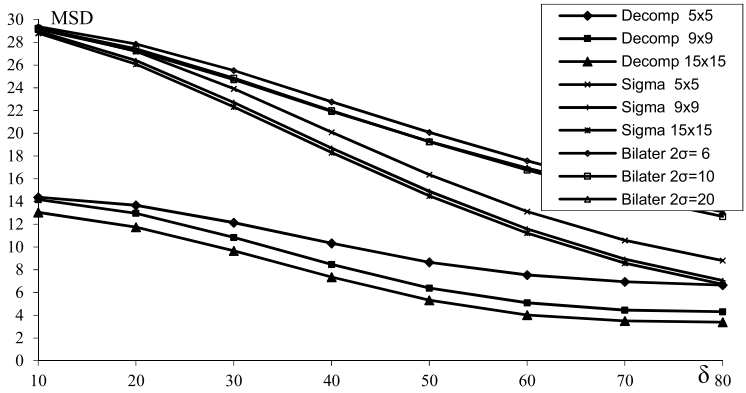


Figure 3.7. Dependences of signal recovery precision (MSD) on the effective window size (δ) for different types of smoothing filter.

are in Figure 3.7. Every type of filter forms its own separate bunch of lines. The lines of the sigma filter and bilateral filter are located near each other and differ by 15–20%. At the same time, the lines of the decomposition filter are located separately and demonstrate 2–3 times more accurate restoration.

The examples in Figure 3.6 and graphs in Figure 3.7 demonstrate that sigma filter characteristics are slightly better than bilateral filter ones, despite the image being the signal, which should demonstrate the maximum difference between the filters. For small-sized objects, comparable with the fragment size, the difference between them will be negligible. This fact may be explained as follows: the shape and size of details, especially small ones, define the spatial area that is factually used in smoothing. In such a case, the size of the spatial analysis window is less significant.

The decomposition algorithm reveals much better characteristics, thus demonstrating the importance of preliminary estimation stage over small-sized vicinity. Note, that the smoothness (more precisely, its inverse value) may be interpreted as the signal complexity. For comparison of smoothing algorithms, the two-dimensional variations (3.19)–(3.21) were used.

The two-dimensional variation estimates of smoothing results are given in Table 3.4. One can see that sigma filter (b) and bilateral filter (c) leave a significant part of noise peaks in Figure 3.6(a) unchanged (large MSD and w_1, small d values). Local mean (e) and median (f)

Table 3.4. The changes of MSD, object number index (w_1), and object size index (d) depending on the smoothing (source image without noise and images *a*-*f* in Figure 3.6).

Estimate/Image	*Source*	*a*	*b*	*c*	*d*	*e*	*f*
MSD (scales)	0.00	29.84	6.37	8.71	3.03	14.15	10.93
w_1	0.39	380.8	14.67	33.93	0.43	1.37	1.70
d	22.5	1.41	1.84	1.42	25.11	9.85	8.82



Figure 3.8. Smoothed components (s) of images in Figure 3.1 after decomposition (3.33).

demonstrate well smoothing (small w_1 and large d), but the signal noticeably differs from the original one (large MSD). The decomposition algorithm (d) demonstrates the best results in all parameters, which are close to undistorted values.

3.5.3.2 *Experiments with real images*

The images in Figure 3.1 (256×256 pixels) were smoothed by decomposition algorithm (3.33) with smoothing window 15×15 pixels; results are presented in Figure 3.8. The values of object number index (w_1), second Kronrod variation (w_2), object size index (d), and entropy (E) are shown in Table 3.5 both for source and smoothed data.

Table 3.5. Object number index (w_1), second Kronrod variation (w_2), object size index (d), and entropy (E) for images in Figure 3.1 and smoothed components (s) in Figure 3.6.

	Source image (Figure 3.1)				Smoothed component s (Figure 3.6)			
Image	w_1	w_2	d	E	w_1	w_2	d	E
a	469.85	5260.25	2.80	7.59	148.66	3272.62	5.50	7.44
b	129.23	2589.58	5.01	7.60	15.12	1476.61	24.42	7.48
c	618.79	7947.70	3.21	7.59	276.97	5516.53	4.98	7.48
d	360.03	4804.38	3.16	7.16	110.68	2597.77	5.87	7.03
e	256.84	3260.76	3.17	7.34	70.27	2838.65	10.10	7.17
f	419.62	5118.65	3.05	6.76	51.33	2703.48	13.17	5.93
g	231.24	6462.04	6.39	5.61	65.40	5258.05	20.10	4.73

Comparing the data in Table 3.5 for source images and their smoothed components (s), one can note the following:

(i) The object number indices (w_1) appreciably reduce 3.5–9 times.
(ii) The object size indices (d) increase for all images 1.5–5 times.
(iii) The second variations (w_2) are slightly reduced 1.2–1.9 times.
(iv) The entropy values (E) change tiny and are few informative.

Analyzing the data in Table 3.5 concerning smoothed component, we can disclose two different groups of values: The first one includes "a", "c", and "d" images, and the second one contains "b", "e", "f", and "g" ones. The first group demonstrates large object number indices (w_1) and small object size (d) indices. In the second group, vice versa, values w_1 are small, and values d are large. One can see that for the first group, the values w_1 decrease 2.2–3.2 times, and values d decrease 1.5–2.0 times only. For the second group, these ranges are noticeably differing: 3.5–8.6 and 3.2–4.9.

Let us use the complexity estimate explanations that are done in Section 3.4. Large w_1 and small d for smoothed components of "a", "c", and "d" images denote that they contain many small objects. Small w_1 and large d for other images indicate that they have a relatively small number of large-sized details. Comparing w_1 and d values, one can say that the first group images are more complicated than the second group ones.

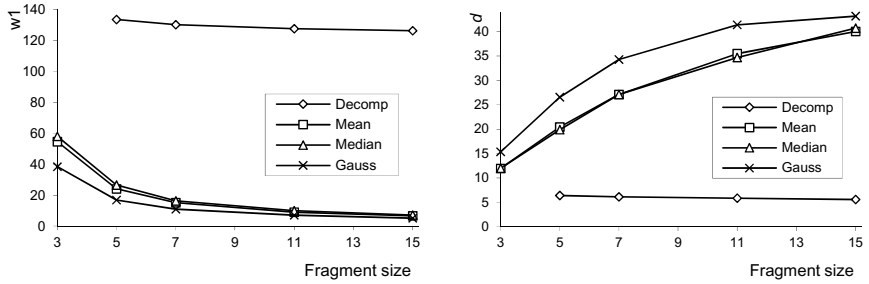


Figure 3.9. The dependences of object number (w_1) and object size (d) indices on the size of the analysis fragment for the images in Figure 3.1(a) for different smoothing algorithms.

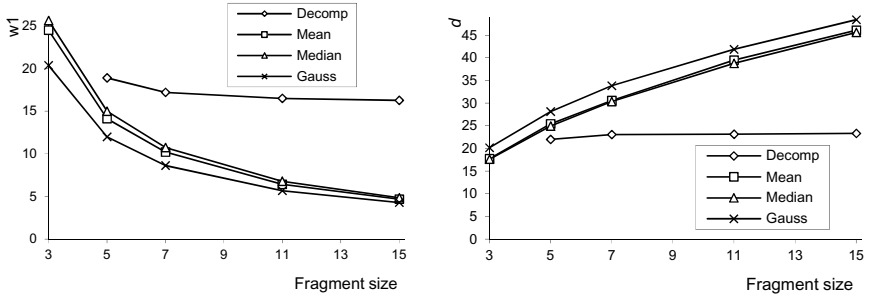


Figure 3.10. The dependences of object number (w_1) and object size (d) indices on the size of the analysis fragment for the images in Figure 3.1(b) for different smoothing algorithms.

Figures 3.9–3.11 demonstrate the dependences of the object number (w_1) and the object size (d) indices on the analysis window size; the data correspond to images $a - c$ in Figure 3.1. The following smoothing algorithms were tested: *mean*, *median*, *Gauss*, and *decomposition*.

In spite of the differences in absolute values, the behavior of corresponding plots is the same and coincides with the dependencies in Figures 3.4 and 3.5 for test images. So, image smoothing with mean, median, and gauss filter demonstrates the tendency of reducing the object number index (w_1) to zero and increasing the object size index (d), while the fragment size enlarges. Such behavior of w_1 and d values indicates the destruction of image details.

By contrast, if an image was smoothed by a decomposition algorithm, the values w_1 and d demonstrate stabilization of the results

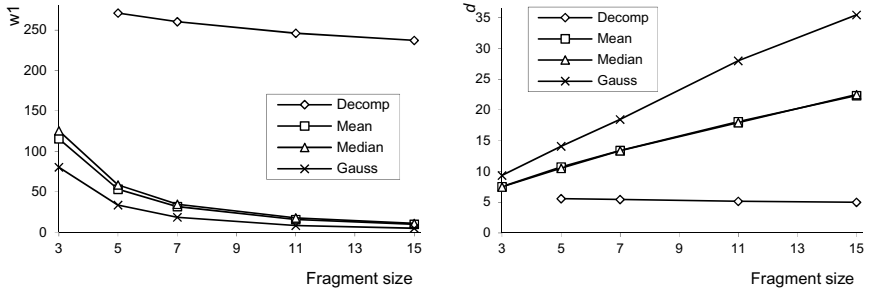


Figure 3.11. The dependences of object number (w_1) and object size (d) indices on the size of the analysis fragment for the images in Figure 3.1(c) for different smoothing algorithms.

and indicate the preservation of image details. The resulting values of the object number index (w_1) and the object size index (d) constitute the complexity estimate for the image smoothing component.

3.5.4 *Color (multispectral) image decomposition*

According to model (3.11), the value of color (multispectral) image should is considered as a vector $\mathbf{x}_{mn}$ in Z-dimensional spectral space:

$$\mathbf{x}_{mn} = \mathbf{S}_{mn} + \mathbf{t}_{mn}, \tag{3.34}$$

where $\mathbf{S}_{mn}$ is the value of the multidimensional smoothed component at the point (m, n) and $\mathbf{t}_{mn}$ is the difference vector, containing the brightness-color information about the texture, fine details, and noise.

The variety of possible color image pixel values composes the color cube of K^Z volume. The local histogram over the fragment W_{mn} for the single-channel image is defined by formula (3.26). For the multispectral image, it is represented as follows:

$$h_{mn}(k_1, \ldots, k_Z) = P(x = \{k_1, \ldots, k_Z\} | x \in W_{mn});$$
$$\sum_{k_1=0}^{K-1} \ldots \sum_{k_Z=0}^{K-1} h_{mn}(k_1, \ldots, k_Z) = 1. \tag{3.35}$$

A point $h(k_1, \ldots, k_Z)$ of the multispectral histogram $H(W_{mn})$ is the probability that an element x_{ij} of the fragment W_{mn} has the value

$x_{ij} = \{k_1, \ldots, k_Z\}$. The concentration of non-zero points is analog of one-dimensional histogram $h(k)$ modes that respond to the portions of Q areas $U^1, \ldots, U^Q$, covered by the fragment W_{mn}. The concentration centers are the mean vectors $\mathbf{S}^1, \ldots, \mathbf{S}^Q$.

Color image decomposition is the determination of voxel vectors $\mathbf{S}_{mn}$. This is similar to finding the S_{mn} component of a gray-scale image, considered in Section 3.5.1.1. It consists in selecting the concentration in the histogram $H(W_{mn})$ for the central region of W_{mn} and finding its mean vector $\mathbf{S}^1$, which is considered as the smoothed value $\bar{\mathbf{x}}_{mn}$ at the point (m, n).

The algorithm for the decomposition of Z-channel image $\mathbf{X}$ and determining the values $\bar{\mathbf{x}}_{mn}$ may be constructed on the scheme, similar to the decomposition algorithm (3.33) for gray-scale images. The difference is that except for the brightness analysis intervals Δ^V and Δ^W, the *hypersphere* of the corresponding radius, centered at $\bar{\mathbf{x}}_{mn}$ point, should be applied. Also, in (3.33), the average of the vector set should be used because the median estimate is not directly applicable to the volumetric set of vectors:

$$\mathbf{s}_{mn} = \mathbf{A}(W_{mn}, \bar{\mathbf{x}}_{mn}, n^W, \Delta^W). \tag{3.36}$$

For some kinds of estimates, the multidimensional analog may be proposed. In supposition of the isotropy of Z-dimensional space, the norm of the vector difference should be used instead of the modulo of brightness difference. The expression for calculating the local average (3.28) will be

$$\bar{\mathbf{x}}_{mn} = \sum_{z=1}^{Z} \mathbf{x}_z w(\|\mathbf{x}_z - \mathbf{x}_0\|) \Bigg/ \sum_{z=1}^{Z} w(\|\mathbf{x}_z - \mathbf{x}_0\|). \tag{3.37}$$

The sigma filter (3.29) in the multidimensional form is described as

$$\bar{\mathbf{x}}_{mn} = \mathbf{A}(V'_{mn}, \mathbf{x}_{mn}, \delta) = \sum_{r=1}^{R} \mathbf{x}^r_{mn}/R \quad \text{for } \|\mathbf{x}^r_{mn} - \mathbf{x}_{mn}\| \le \delta. \tag{3.38}$$

The norm of vector difference can be interpreted as the distance between two points in multidimensional space.

Practical realization of color image decomposition algorithm needs large computational resources because it assumes the analysis

of Z-dimensional histogram in every image point. To simplify the calculations, component-wise estimation may be used. This means the projection of multidimensional distribution $H(W_{mn})$ to coordinate axes and separate analysis of Z one-dimensional histograms. Such projection may lead to overlapping of two or more concentrations, which cannot be separated later. Nevertheless, such an event is enough rare and this approach usually produces suitable results.

3.5.5 *Image decomposition summary*

The obtained results allow us to draw the following conclusions:

(i) Within the framework of a two-scale image model, decomposition corresponds to the division of the image into different information components: a piecewise-smooth component, carrying the information about the shape and brightness of lengthy objects, and a texture-detail component, containing fine details, texture, and noise.

(ii) The introduction of locality in the domain of values, in fact, means the spatial restriction of the domain of the element (as a subset of the analysis) according to the shape of the object that contains the processed element.

(iii) As the size of the analysis area of the decomposition algorithm increases, the signal restoration error tends to zero, but the image complexity decreases only to some level. This indicates that the details of the image are preserved. When the local mean or local median is used, the error increases almost linearly, and the image complexity tends to zero. This indicates the destruction of image details.

(iv) The decomposition algorithm provides better recovery of the noisy signal than other smoothing methods.

(v) On the basis of the two-scale image model and decomposition method, allowing to divide an image into components with different information content, a new approach to image analysis and processing is possible. It lies on the concept that for the analysis one should select only that component of the image, which should contain the required information. At the processing, only the transformations specific for every component should be used.

3.6 Topological Characteristics and Local Image Analysis

Essential drawback of direct use of the 2D variations is that they are *global* estimates of a function. Therefore, this fact involves difficulties when some problem requires *local* function analysis. We investigate the construction of metric characteristics on the base of two-dimensional variations. The application of these features is explored for analysis of local topological image features and for detecting the objects and noise. The conceptions of the size and amplitude indices of object convexity are introduced [7]. The approach for their composition is proposed, and their applications in algorithms for objects and noise detection are illustrated. The theoretical conclusions are experimentally verified.

3.6.1 *Formation of topological characteristics*

Let continuous function $f(x, y)$ be specified on two-dimensional variety D and have a finite number of components of the set e_t in each of its t-levels, $t\chi(0, T)$ (see Section 3.3.3). Assume that t-level set of this function e_t contains $v_0(e_t)$ non-intersecting components. Let us consider inner regions $c_t(i)$ of every i-component in the t-level: $c_t(i)$, $1 \le i \le v_0(e_t)$.

We introduce the *convexity index* parameter $z(c_t(i))$ for the region $c_t(i)$ as the value $z(c_t(i)) = z_t(x, y)|(x, y) \in c_t(i)$ on the assumption that

$$z_t(x, y|(x, y) \in c_t(i)) = 1, \quad \text{if } (f(x, y) - t) \ge 0 \quad \text{and}$$
$$z_t(x, y|(x, y) \in c_t(i)) = 0, \quad \text{if } (f(x, y) - t) < 0.$$

Let function $a(c_t(i)) \ge 0$ reflect the *size* of the region $c_t(i)$ and $a_t(x, y) = a(c_t(i))$ for $(x, y) \in c_t(i)$. Note that "*size*" may be one of different kinds of region estimates, for example, an area, border length, or diameter of the figure. Introduce some nonnegative *function of sizes* $s(a) \ge 0$, and construct the following functions $p_t(x, y)$ and $n_t(x, y)$ in D:

$$p_t(x, y) = z_t(x, y)s(a_t(x, y)) \quad \text{and}$$
$$n_t(x, y) = (1 - z_t(x, y))s(a_t(x, y)). \tag{3.39}$$

As one can see, $p_t(x,y) > 0$ for the components, where $f(x,y)$ is convex upwards, and $p_t(x,y) = 0$ for the components, where $f(x,y)$ is convex downwards. On the contrary, $n_t(x,y) > 0$ for the components, where $f(x,y)$ is convex downwards, and $n_t(x,y) = 0$ for the components, where $f(x,y)$ is convex upwards. Integrating the functions $p_t(x,y)$ and $n_t(x,y)$ on t, let us define the *convexity indices upward* and *downward* at each point $(x,y) \in D$ as

$$p(x,y) = \int_0^\infty p_t(x,y)dt \quad \text{and} \quad n(x,y) = \int_0^\infty n_t(x,y)dt. \tag{3.40}$$

Or, for the discrete case,

$$p(x,y) = \sum_{t=0}^{T-1} p_t(x,y)/T = \sum_{t=0}^{T-1} z_t(x,y)s(a_t(x,y))/T \quad \text{and}$$

$$n(x,y) = \sum_{t=0}^{T-1} n_t(x,y)/T = \sum_{t=0}^{T-1} (1 - z_t(x,y))s(a_t(x,y))/T. \tag{3.41}$$

The functions $p(x,y)$ and $n(x,y)$ are, *per se*, *metric characteristics* of local topological features of analyzed function $f(x,y)$, based on the function of sizes $s(a)$. The obtained functions can be named as follows: $p(x,y)$ as the *size and amplitude index of upward convexity* and $n(x,y)$ as the *size and amplitude index of downward convexity.*

It is important to note that in contrast to two-dimensional functionals (3.18)–(3.20), which are *global* characteristics of a function $f(x,y)$, the functions $p(x,y)$ and $n(x,y)$ are *local* characteristics of the function and their values are related to every coordinate point (x,y). Therefore, these estimates can be used for the analysis of *local image features*, in particular, for detecting the objects and the noise [7, 12, 13].

3.6.2 *Detection of the noise and objects*

3.6.2.1 *Definition of the pulse noise*

Various types of noises can distort real images. Many types of research were assigned to the problem of their detection and removal; the overall question goes far beyond the border of this work. Now

we limit our consideration only to the task of detecting the pulse noise [6, 12, 55].

The model of image distortion with pulse noise is enough simple [6, 12]. The value of each image element $f(x, y)$ accidentally with the probability p is replaced by random value $\xi(x, y)$. The distortion function is as follows:

$$f(x,y) = \begin{cases} f_0(x,y) & \text{with probability } (1-p) \text{ for undistorted pixel;} \\ \xi(x,y) & \text{with probability } p \text{ for distorted pixel.} \end{cases} \tag{3.42}$$

Let us consider that pulse noise values $\xi(x, y)$ are distributed uniformly in the range $[0, T-1]$. As it was noted in [6], the task of pulse noise filtering consists of two stages: the detection of distorted elements and the correction of their values. The most difficult and crucial is the noise detection stage.

Note that the pulse noise appears in a t-level set e_t as independent components of one-pixel size, therefore, they may be considered as the components with minimal possible dimension and perimeter.

3.6.2.2 *Object and noise detection algorithm*

The function of sizes $s(a)$ in (3.39)–(3.41) plays a key role in the algorithm for detecting the objects or pulse noise. The shape of this function allows controlling the selection of detected objects depending on their dimensions.

The simplest function of sizes $s(a)$ is threshold splitting of the range of possible values to non-zero and zero elements. For example,

$$s(a) = 1, \quad \text{if } a \le r, \quad \text{and} \quad s(a) = 0 \text{ otherwise.} \tag{3.43}$$

The choice of $r = 1$ means the pulse noise detection algorithm.

More general is the three-range variant of $s(a)$ function:

$$\begin{aligned} &s(a) = 1, \quad \text{if } r_1 \le a \le r_2, \quad \text{and} \\ &s(a) = 0, \quad \text{if } a < r_1 \quad \text{or} \quad a > r_2. \end{aligned} \tag{3.44}$$

The boundaries $[r_1, r_2]$ define the range of object sizes to be detected.

The choice of r_1 and r_2 values and the threshold for clipping $p(x, y)$ and $n(x, y)$ functions allow constructing the indicator function for detecting the noise and the objects. Of course, it is possible

to build a more complicated function of sizes $s(a)$, for example, multirange ones.

3.6.2.3 *Application to color and multispectral images*

The solution of the object detection problem in gray-scale images can be expanded to color and multispectral images. Two variants of application of the discussed above results are available:

(i) Compose the most informative image component, for example, the first term in principal components decomposition (Hotelling transformation) [19,22] or the brightness (or some another) component of appropriate multispectral image presentation. According to the given function of sizes, $s(a)$ in (3.44) mark the necessary objects (regions) in this component.
(ii) Detect the objects in every component of the multispectral image separately, using only the bottom limitation in size (r_1) of function $s(a)$ in (3.44). Combine the boundaries, obtained for all components, to a common map of regions. Remove from this map all the areas whose sizes are less than r_1 or greater than r_2 in (3.44).

3.6.3 *Detection examples*

3.6.3.1 *Pulse noise detection*

Figure 3.12 demonstrates an example of noise detection. Image (a) was distorted by pulse noise (3.42) with probability $p = 0.03$; MSD



Figure 3.12. Pulse noise detection: (a) distorted image (noise 3%); (b) detected noisy pixels restored; (c) the 4-times contrasted difference between (b) and not distorted image.

of the distorted image from the original one equals 13.99 grayscales. Image (b) is the result of the restoration of the pixels, detected in (a) by the algorithm (3.43) with $r = 1$. MSD of the restored image from the original one equals 0.87 grayscales. Image (c) represents a 4-times contrasted difference between restored (b) and original images.

In image (c), one can see only a small portion of undetected noisy pixels. The first reason that the algorithm (3.43) did not detect these pixels is the following. Due to the randomness of noisy pixels' location, some of them can be neighbors that create a group. If the values ξ (in 3.42) of noise pixels in a group are close, these pixels constitute some false object. Its area exceeds the area limit, and therefore this object will be detected. Such defects can be avoided with the iterative process when after the first noise detection step with $r = 1$, the second iteration accomplishes with $r = 2$ or more. The other reason is that some bugs tightly adjoin the sharp edge of lengthy objects. If the difference between the bug and the object values is small, then the bug, *per se*, spatially joins to this object, distorting its shape. In such a case, this bag cannot be detected without morphological analysis.

3.6.3.2 *Object detection*

Based on a three-range function $s(a)$ in (3.44), it is possible to construct the area-based object detection algorithm. In Figure 3.13, examples of object detection with different ranges (r_1 and r_2) of the function $s(a)$ are demonstrated.

Image (a) is a source air photo (512×512 pixels) and (b)–(f) are indicator functions of detected objects. Small-sized objects (range 20–80 pixels) with downward convexity are in image (b), and objects of the same range with upward convexity are in (c). Middle-sized objects (1000–3000 pixels) are, analogously, in images (d) and (e). Large-sized objects of downward convexity (range 12000–20000 pixels) are in image (f).

3.6.3.3 *Object detection in color image*

Figure 3.14 demonstrates the example of object detection in a color image. The map of spots on a giraffe's skin is unique, and the problem to solve is the formation of a catalog of animal spot patterns for the identification of individuals. The discussed above object detection

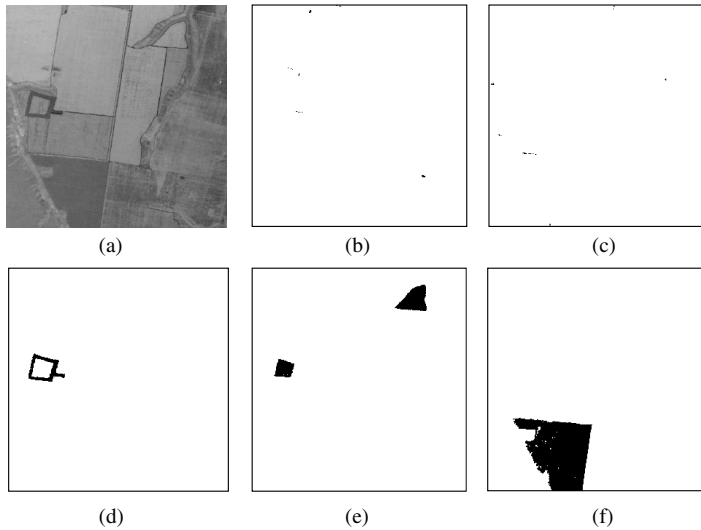


Figure 3.13. Detection of objects basing on their area: (a) source image; ((b), (c)) small range (20–80) downward and upward convexities; ((d), (e)) middle range (1000–3000) downward and upward convexities; (f) large range (12000–20000) downward convexity.

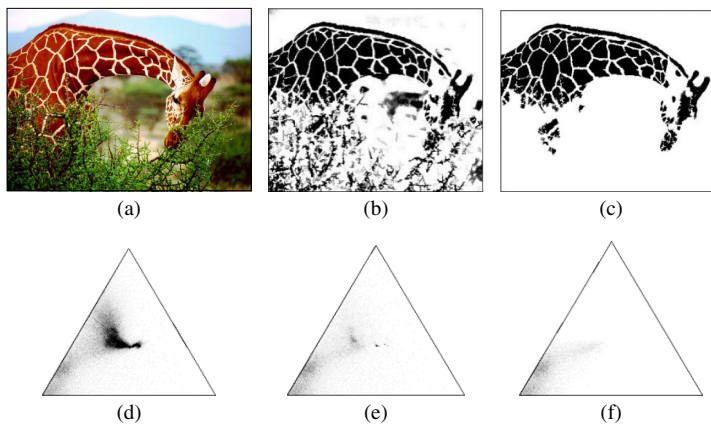


Figure 3.14. Object detection in color image: (a) brightness component; (b) detected areas with downward convexities in the range of 25–2000 pixels; (c) the result of separation; (d), (e), and (f) are corresponding chromaticity–saturation distributions for the images (a)–(c).

method can be used for it. Figure 3.14(a) is the brightness of the source color image (from the Adobe album); in (b) are the downward convexity areas in the range of 25–2000 pixels.

The later separation of detected areas was realized using the probability distribution of image elements dependent on their chromaticity-saturation parameters. These distributions in Maxwell triangle space [19, 46] for the images (a)–(c) in Figure 3.14 are presented in images (d)–(f). The power of darkness in (d)–(f) is proportional to the probability of image pixels with corresponding chromaticity-saturation values. The separation was realized by localizing in distribution (e) the region, corresponding to average color parameters of desired spots, and by removing from the image (b) those areas, in which color mapping of average value exceeds the boundaries of the established color region. Image (f) is the resulting distribution for residuary areas. Figure 3.14(c) demonstrates the final set of detected areas after the separation of corresponding color characteristics.

3.7 Distortion Diagnostics and Image Restoration

The processes of image formation and registration can be suffered by various distortions. They may be caused by several reasons: relative movement of an object and registration device, aberrations and diffraction in the optical system, atmosphere turbulence, light dispersion, and so on. When an image is formed with distortions, it may be hardly interpreted and require preprocessing. The way is the decision of inverse task, which is the restoration of an undistorted image based on obtained signal and suppositions about the distortion. There is a huge number of research, devoted to image restoration methods, that use probabilistic and determinative approaches to the decision as empirical treatments and algorithms [19, 37, 47, 52].

The most important stage in the decision of image restoration problem is the determination of the type and parameters of the distortion process, which usually are not known *a priori*. Therefore, the "diagnostics" of the distorting operator by analysis of the obtained image is the actual task.

Note that the relations given in the following correspond to monochrome images, but they can also be applied to color (multispectral) images, considered as a combination of several monochrome components.

3.7.1 *Image formation models and distortion types*

The following linear integral equation may be a sufficient mathematical model for many physical processes:

$$u(x,y) = Ag(x,y) + \xi(x,y), (x,y) \in W, \tag{3.45}$$

where $g(x,y)$ and $u(x,y)$ are the source and observed distorted images, A is the linear integral operator, and $\xi(x,y)$ is additive noise. The image restoration problem is formulated as a finding of some "ideal" image $g(x,y)$, being connected with observed one $u(x,y)$ through the relation (3.45).

The obtained signal $u(x,y)$ can be interpreted as the output of some image formation process, realized by an imaging system, which (in a linear case) is defined by the following equation, similar to (3.45):

$$u(x,y) = s\{A[g]\} + \xi = s\left\{\iint_W h(x,y,\tilde{x},\tilde{y})g(\tilde{x},\tilde{y})d\tilde{x}d\tilde{y}\right\} + \xi(x,y), \tag{3.46}$$

where $g(x,y)$ is the ideal image of the observed scene, $h(x,y,\tilde{x},\tilde{y})$ is the point spread function of the distorting system, function $s\{\cdot\}$ defines sensor nonlinearity, and ξ is an uncorrelated additive random noise.

If the impulse response is stationary in the image area, $h(x,y,\tilde{x},\tilde{y})$ will be a function of the differences of arguments: $h(x-\tilde{x}, y-\tilde{y})$. Then, (3.46) becomes a more familiar convolution integral:

$$u(x,y) = s\left\{\iint_W h(x-\tilde{x},y-\tilde{y})g(\tilde{x},\tilde{y})d\tilde{x}d\tilde{y}\right\} + \xi(x,y). \tag{3.47}$$

3.7.1.1 *Generalized distortion models*

Motion blur

Motion blur appears as the result of the relative motion of the object and the camera. It may be the shift, rotation, scaling, or their combination. The general mathematical model of a blur can be written as

$$u(x,y) = \int_0^T J_f(\tilde{x},\tilde{y},t)g(f_1(\tilde{x},\tilde{y},t),f_2(\tilde{x},\tilde{y},t))dt, \tag{3.48}$$

where $J(x, y, and\, t)$ is Jacobian of the motion:

$$J_f(\tilde{x},\tilde{y},t) = \begin{vmatrix} \partial f_1/\partial\tilde{x} & \partial f_1/\partial\tilde{y} \\ \partial f_2/\partial\tilde{x} & \partial f_2/\partial\tilde{y} \end{vmatrix}.$$

The simplest case is 1D distortion: the translation along one axis (say, horizontal one) with the constant speed V along with the interval $[0, T]$. The point-spread function is

$$h(x-\tilde{x}) = \begin{cases} 1/VT, & 0 \le x-\tilde{x} \le VT; \\ 0, & \text{otherwise.} \end{cases} \tag{3.49}$$

For uniform accelerated motion $\tilde{x} = (x - at^2)$, $a > 0$, $0 \le t \le T$. The point-spread function will be

$$h(x-\tilde{x}) = \begin{cases} 1/2a(x-\tilde{x}), & x - aT^2 \le \tilde{x} \le x; \\ 0, & \text{otherwise.} \end{cases}. \tag{3.50}$$

The uniform rotation blur can be determined in polar coordinates as

$$h(\theta,\varphi) = \begin{cases} 1/|\omega|, & \theta_0 \le \theta - \varphi \le \theta_0 + \omega T; \\ 0 & \text{otherwise.} \end{cases}. \tag{3.51}$$

Optical aberrations

Optical aberrations are the frequent disturbances in image acquisition. In the following, we consider the defocusing, which is the

simplest aberration type. To obtain a complete defocusing model, one needs to know the intensity distribution within the spread spot, caused by a point object.

From the geometrical optics, it is known that the intensity distribution should be roughly constant, non-zero within the spread circle, and zero elsewhere. This corresponds to the following point-spread function:

$$h(x,y,\tilde{x},\tilde{y}) = \begin{cases} 1/(\pi r^2), & (x-\tilde{x})^2 + (y-\tilde{y})^2 \le r^2; \\ 0, & \text{elsewhere}, \end{cases} \tag{3.52}$$

where r is the spread circle radius.

Atmospheric scattering

A detailed consideration of the point-spread function and other relevant properties for atmospheric scattering and turbulence can be found in special literature, like in [48]. For a Kolmogorov turbulence spectrum [28], the average point-spread function is rotationally symmetric:

$$h(x,y,\tilde{x},\tilde{y}) = \exp\{-3.44(\lambda|r|/r_0)^{5/3}\}, \tag{3.53}$$

where $r = \{(x-\tilde{x})^2 + (x-\tilde{y})^2\}^{1/2}$, $r_0 \approx \lambda^{6/5}(\cos\gamma)^{3/5}$ is the diameter of the diffraction-limited lens, considering that the area of its Airy disk equals the observation aperture, and γ is the zenith angle. In order to illustrate the form of the turbulence point-spread function, the atmosphere turbulence is typically described by a Gaussian function:

$$h(x,y,\tilde{x},\tilde{y}) = C\exp\{-((x-\tilde{x})^2 + (y-\tilde{y})^2)/2\sigma^2\}. \tag{3.54}$$

Non-optical imaging technique distortions

A number of different image formation techniques are available, based on measuring non-optical radiations, like X-ray and gamma-ray imaging, sonar imaging, seismic vibrations, *et al.* As usual, the models of non-light imaging are multiple factor and use additional suppositions essentially complicating the task. Generally speaking, these models are outside of the framework of this book. Nevertheless, some kinds of distortions (for example, blur due to relative motion) may be interpreted as optical ones; in that case, the relations considered above can be used.

3.7.2 *General approaches to image restoration*

The restoration task is a typical inverse problem with incomplete data. The approaches to its decision are typically based on the decision of an ill-posed problem [52] or on statistical estimation methods [47]. In the first case, the procedure reduces to the minimization of the following Tikhonov functional [52] with residual ρ in metric μ:

$$g^* = \arg\min_{g \in G} \{\rho_\mu(V) + \alpha\Omega(g)\}. \tag{3.55}$$

In the second case, when the Bayesian approach or *a posteriori* probability criterion is used, the procedure reduces to the optimal estimation task:

$$g^* = \arg\min_{g \in G} \{-\ln q(Ag - u) - \ln p(g)\}, \tag{3.56}$$

where $p(g)$ and $q(\xi)$ are the probability densities of the image and of the additive noise: $\xi(x, y) = Ag - u$ and g^* is the optimum in image space G.

The image restoration quality depends on the properties of operator A. This operator, describing the image-forming system, as a rule, is irreversible. So, the decision of equation (3.45) will be multivalued even under exactly defined data and unstable with respect to the data errors [52]. These reasons essentially restrict image restoration accuracy under distortions and noise.

In the case of an inverse problem with incomplete data, the result of restoration comes to some compromise between the image sharpness and noise intensification. If on the basis of a distorted image one needs to decide a recognition task, it can be effective to limit in decision to the partial task only, except the decision of the whole image restoration task; for example, to restore only the phases of image spectrum or only some of its components.

3.7.2.1 *Linear methods*

The linear algorithms are used most frequently in image restoration since their implementation is rather easy. Such an algorithm can be represented as the minimization of the quadratic functional:

$$g^* = \arg\min_{g \in G} \{\|Ag - \tilde{u}\|^2 + \alpha\Omega(g)\}. \tag{3.57}$$

The principal factors defining the restoration quality are the properties of distorting operator A and the noise level.

3.7.2.2 *Nonlinear methods*

Noticeably different results can be obtained with nonlinear restoration methods. If we interpret an image as a function of bounded variations [37], the restoration task can be written as the minimization of functional:

$$g^* = \arg\min_{g \in G} \{\|Ag - \tilde{u}\|^2 + \alpha \mathrm{Var}(g)\}. \tag{3.58}$$

A number of different definitions for variations of the function of several variables are known; the most significant of them were considered in Sections 3.3 and 3.4, where the image complexity was discussed. In our opinion, two-dimensional Kronrod variations (3.18) are the most attractive ones. In this case, the regularizing functional (3.58) may be rewritten as

$$g^* = \arg\min_{g \in G} \{\|Ag - \tilde{u}\|^2 + \alpha F(w_1, w_2)\}. \tag{3.59}$$

3.7.2.3 *Fourier-transform method*

The possibilities of simplification of the minimization algorithm depend on the specificity of the distortion operator. Usually, the homogeneous integral transformation is used as the image-forming system model; it leads to the following convolution-type equation:

$$u(x,y) = Ag + \xi = \iint h(x - \tilde{x}, y - \tilde{y}) g(\tilde{x}, \tilde{y}) d\tilde{x} d\tilde{y} + \xi(x,y).$$
$$-\infty < (x,y) < \infty. \tag{3.60}$$

Basedg on the Fourier transform, it comes to the spectral transformation:

$$U(\omega_1, \omega_2) = H(\omega_1, \omega_2) G(\omega_1, \omega_2) + \Xi(\omega_1, \omega_2), \tag{3.61}$$

where U, H, G, Ξ are the Fourier transforms of appropriate functions in (3.60). The linear decision of this equation can be written [37] as

$$\hat{g}(x,y) = \iint K(x - \tilde{x}, y - \tilde{y}) \tilde{u}(\tilde{x}, \tilde{y}) d\tilde{x} d\tilde{y}, \tag{3.62}$$

where K is the inverse transform kernel, which, under certain conditions, is

$$K(x,y) = \frac{1}{4\pi^2} \iint \frac{H^*(\omega_1,\omega_2)e^{i(x\omega_1+y\omega_2)}}{|H(\omega_1,\omega_2)|^2 + \alpha(1+\beta(\omega_1+\omega_2))} d\omega_1 d\omega_2. \tag{3.63}$$

The regularization parameters $0 < \alpha < 1$ and $0 < \beta < 1$ depend on a specific task and the noise level; they may be defined by one or another criterion [52] and expert evaluations. The most number of restoration algorithms, using the Fourier transform, are based on formulas (3.60)–(3.63).

The observed image $u(x,y)$ is normally specified in the bounded area; this disables applying the Fourier transform to equation (3.60) without expanding the distorted image. One of the solutions to this problem for the discrete approximation of equation (3.60) was considered in [17].

3.7.3 *Distorting operator diagnostics*

In practice, we often need to diagnose the type and parameters of distortion, corrupting the observed image. For the moment, this problem is not enough studied. In the following, we limit by the consideration of the linear space-invariant systems, being described by the convolution equation (3.60).

The first step of the image restoration procedure is the identification of distortion type and parameters. If we know the camera characteristics, its relative position, and movement with respect to an object, we can calculate the point-spread function. In practice, these data are rarely known and have to be defined by examining the observed distorted image.

The distortions, described by 2D convolution equation (3.60), can be reduced to the pair of 1D transforms along with x_1 and x_2, definable as

$$u(x) = Ag = \int_{-\infty}^{\infty} h(x-\tilde{x})g(\tilde{x})d\tilde{x}, \quad -\infty < x < \infty. \tag{3.64}$$

This is the fundamental transformation in radiophysics, communication theory, electrotechnology, and others. We can use the analysis

methods, developed in these branches of science. In spectral form, (3.64) will be

$$U(\omega) = H(\omega)G(\omega). \tag{3.65}$$

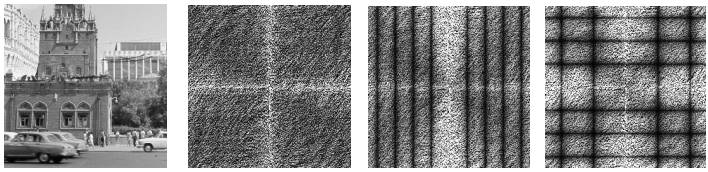
In the general case, source image spectrum $G(\omega)$ does not possess any specific properties. But for distorting operator spectrum $H(\omega)$, and hence for distorted image spectrum $U(\omega)$, we can form some list of properties.

According to model (3.64), the distorting operator A of the size R_A is defined by the kernel of integral transform $h(x)$. In real tasks, $h(x)$ is a function with finite support; hence, $h(x)$ equals zero in $|x| \geq R_A$. At the same time, a real image is a function, defined on the whole plane. This difference, in fact, creates the basis for the following discussion.

It is acceptable to suggest that the spectra of transfer functions $H(\omega)$ essentially differ from the spectra of real images. Over the amplitude spectrum of the real image, one can make only general conclusions concerning image characteristics [19]. As it follows from the experiments, the amplitude spectrum of distorted image retains the distinctive structure of the spectrum of transformation kernel $h(x)$; this is correct for uniform motion blur, defocusing, aberrations, turbulent medium, *et al.* Above all, this is caused by the ordered arrangement of zeroes in the complex plane (ω_1, ω_2).

The properties of spectrum $H(\omega)$ are held for the whole complex plane and, in particular, for its real axis. If a system transfer function reduces to zero in the points $\nu_1, \nu_2, \ldots, \nu_n$ on the real axis, then, according to (3.65), on the base of spectrum $H(\omega)$, it is possible to estimate the distorting operator type, the size of the carrier R_A, and therefore some connected parameters like object speed, defocusing size, etc. As a rule, the kernel of the distorting operator $h(x)$ is even and continuously differentiable except for the set of points with the coordinate $\pm nR_A$, setting the size of the defocusing operator. In these restrictions on $h(x)$, the spectrum $H(\omega)$ has points of zeroes in the real axis.

In practice, the diagnostics of the distorting operator over the violent image can be done by comparing the observed image spectrum with the spectra of etalon distorting operators $H(\omega_1, \omega_2)$. These etalons can be obtained by numerical modeling or by testing specific image-forming systems. In the case of a 1D rectangular



(a) (b) (c) (d)

Figure 3.15. (a) Source image; (b) its amplitude spectrum; (c) spectrum after 1D defocusing of 9 pixels width; (d) spectrum after 2D defocusing by the rectangular aperture of 5×7 pixels.

operator of width u, the $H(\omega)$ spectrum transforms as $H'(\omega) = H(\omega)2u(\sin u\omega)/u\omega$. Therefore, it becomes zero in the points $\omega = n\pi/u$; $n = 1, 2, \ldots$, producing specific periodic structures that can be simply detected in the spectrum (Figures 3.15(c) and 3.15(d)).

Figure 3.15 demonstrates the source image (a), the amplitude component of its spectrum (b), and the spectra after defocusing by the linear aperture of 9 pixels width (c), and by the rectangular aperture of 5×7 pixels (d). For clarity, upper-frequency amplitudes of the spectra in images (b)–(d) are magnified approximately up to uniform.

After the defocusing by image shift along some direction, $h(\omega_1, \omega_2)$ spectrum will contain the strips with close to zero values, oriented in the direction, perpendicular to the shift (Figures 3.15(c) and 3.15(d)). The distances between these strips are equal to L/u, where L is the image size and u is the shift value. These parameters indicate the size and orientation of the defocusing aperture.

It is easy to construct an algorithm for detecting the direction and the distances between such strips. The algorithm can calculate the variations $\mathrm{Var}(\alpha)$ of mean values, being found along with the variety of secant lines, parallel to every direction α_i, $0 \le \alpha_i < 2\pi$. Defocusing directions β_i will be the directions, perpendicular to the local maxima of $\mathrm{Var}(\alpha_i)$.

The possibility of experimental diagnostics of distorting operator was confirmed with the recognition system [3]; the recognition procedure and obtained results were published in [2, 4]. It was demonstrated that the average level of successful definition of the parameters for four types of noiseless distortion varies from 93% for Gauss distribution to 98% for linear motion. Successful identification of the distorting operator type without noise comes to 100%.

In the case of additive normal noise of 0.1%, successful recognition was reached in 78% of experiments.

References

[1] Adams, C.R. and Clarkson, J.A. (1933) On definitions of bounded variation for functions of two variables. *Transactions of the American Mathematical Society*, 36, 824.

[2] Aizenberg, I., Aizenberg, N., and Vandewalle J. (2000) *Multi-Valued and Universal Binary Neurons: Theory, Learning, and Applications* (Springer, Dordrecht).

[3] Aizenberg, I., Aizenberg, N., Butakoff, C., and Farberov, E. (2000) Image recognition on the neural network based on multi-valued neurons. *Proceedings of International Conference on Pattern Recognition*, Vol. 2 (IEEE Computer Society Press, Barcelona), pp. 993–996.

[4] Aizenberg, I., Butakoff, C., Karnauhov, V., Merzleakov, N., and Milyukova, O. (2002) Blurred image restoration using the type of blur and blur parameters identification on the neural network. *Proceedings of SPIE: Image Processing: Algorithms and Systems*, Vol. 4667, pp. 460–471.

[5] Bellettini, G., Caselles, V., and Novaga, M. (2002) The total variation flow in R^N. *Journal of Differential Equations*, 184, 475–525.

[6] Chochia, P.A. (1984) Digital filtering of puls noise in television images. *Technika Sredstv Svjazi: Tekhnika Televidenija*, 1, 26–36 (in Russian).

[7] Chochia, P.A. (2017) Formation of the image topological characteristics based on two-dimensional variations and their application for object and noise detection. *Journal of Communications Technology and Electronics*, 62(12), 1477–1483.

[8] Chochia, P.A. (1998) Image decomposition and enhancement using rank filtering. *Computer and Holographic Optics and Image Processing. Proceedings SPIE*, 3348, 261–266.

[9] Chochia, P.A. (1988) Image enhancement using sliding histograms. *Computer Vision Graphics Image Processing*, 44(2), 211–229.

[10] Chochia, P.A. (1986) Image processing and analysis basing on two-scale image model. *Preprint IITP AS USSR* (VINITI, Moscow) (in Russian).

[11] Chochia, P.A. (1988) Image smoothing on contour preservation. *Image Coding, and Processing* (Nauka, Moscow), 87–98 (in Russian).

[12] Chochia, P.A. (2017) *Methods of Video Information Processing Basing on Two-Scale Image Model* (LAP Lambert Academic Publishing, Saarbrucken) (in Russian).
[13] Chochia, P.A. (2014) Some algorithms for object detection basing on two-scale image model. *Information Processes*, 14(2), 117–136 (in Russian).
[14] Chochia, P.A. (2015) Two-dimensional variation as an instrument for image complexity estimation. *Computer Education and Data Analysis*, 1(11), 1660–1676 (in Russian).
[15] Chochia, P.A. (1988) Two-scale image model. *Image Coding and Processing* (Nauka, Moscow), 69–87 (in Russian).
[16] Chochia, P.A. and Milukova, O.P. (2015) Comparison of two-dimensional variations in the context of the digital image complexity assessment. *Journal of Communications Technology and Electronics*, 60(12), 1432–1440.
[17] Dainty, J.C. (1975) Stellar speckle interferometry. *Laser Speckle, and Related Phenomena*. Topics in Applied Physics, Vol. 9 (Springer), pp. 255–280.
[18] Duda, R.O. and Hart, P.E. (1973) *Pattern Classification and Scene Analysis* (Wiley, New York).
[19] Gonzalez, R. and Woods, R. (2008) *Digital Image Processing* (Prentice Hall, Upper Saddle River).
[20] Haddad, A. and Meyer, Y. (2007) An improvement of Rudin–Osher–Fatemi model. *Applied Computational Harmonic Analysis*, 22, 319–334.
[21] Haralick, R.M. and Watson, L. (1981) A facet model for image data. *Computer Graphics and Image Processing*, 15(2), 113–129.
[22] Hotelling, H. (1933) Analysis of a complex of statistical variables into principal components. *Journal of Educational Psychology*, 24, 417–441, 498–520.
[23] Ivanov, L.D. (1975) *Variations of Sets and Functions* (Nauka, Moscow) (in Russian).
[24] Jain, A.K. (1980) Advances in mathematical models for image processing. *Proceedings of the IEEE*, 5, 502–528.
[25] Jaiswal, R. and Awasthi, A. (2016) Rudin-Osher-Fatemi (ROF) model for blurring removal in digital images. *International Journal for Research in Applied Science & Engineering Technology*, 4, 248–252.
[26] Keelan, B.W. (2002) *Handbook of Image Quality* (Marcel Dekker, New York).

[27] Kharkevich, A.A. (1962) *Spectra and Analysis* (Fizmatgiz, Moscow) (in Russian).
[28] Kolmogorov, A.N. (1965) Three approaches to definition of the "information quantity" concept. *Problemy Peredachi Informatsii*, 1(1), 3–11 (in Russian).
[29] Kronrod, A.S. (1950) On functions of two variables. *Uspekhi Matematicheskikh Nauk*, 5(1), 24–134 (in Russian).
[30] Lagutin, M.B. (2007) *Obvious Mathematical Statistic* (School-book). (Binom, Moscow) (in Russian).
[31] Lee, J.-S. (1980) Digital image enhancement and noise filtering by use of local statistics. *IEEE Transactions on PAMI*, 2, 165–168.
[32] Lee, J.-S. (1983) Digital image smoothing and the sigma filter. *Computer Vision, Graphics, Image Processing*, 24(2), 255–269.
[33] Marr, D. and Hildreth, E. (1980) Theory of edge detection. *Proceedings of the Royal Society of London. Series B*, 207, 187–217.
[34] Mastin, G.A. (1985) Adaptive filters for digital image noise smoothing: An evaluation. *Computer Vision, Graphics, Image Processing*, 31(1), 103–121.
[35] *Mathematical Encyclopedia* (1977–1982) (Soviet Entsiklopediya, Moscow) Vols. 1–5.
[36] Meyer, Y. (2001) Oscillating patterns in image processing and in some nonlinear evolution equations. *The Fifteenth Dean Jacqueline B. Lewis Memorial Lectures, University Lectures Series*, Vol. 22. (American Mathematical Society, Providence).
[37] Milyukova, O.P. (1995) A priory information in reconstruction tasks. *Computer Optic*, 14–15, Part 1, 148–155 (in Russian).
[38] Milyukova, O.P. (1989) Image as a function with limited total variation. *Iconics. Digital Processing of Video Information* (Nauka, Moscow), pp. 19–25 (in Russian).
[39] Milyukova, O.P. and Chochia, P.A. (2013) On estimation of the image complexity by two-dimensional variations. *Journal of Communications Technology and Electronics*, 58(6), 628–635 (in Russian).
[40] Milyukova, O., Kober, V., and Karnaukhov, V. (2012) Chapter 9: Image restoration using two-dimensional variations. *Image Restoration — Recent Advances and Applications* (InTech Publishing, Rijeka), pp. 185–200.
[41] Nagao, M. and Matsuyama, T. (1979) Edge preserving smoothing. *Computer Graphics Image Processing*, 9(4), 394–407.
[42] Nishikawa, S., Massa, R.J., and Mott-Smith, J.C. (1965) Area properties of television pictures. *IEEE Transactions IT-11*, 3, 348–352.
[43] Parzen, E. (1962) On the estimation of a probability density function and mode. *Annals of Mathematical Statistics*, 33, 1065–1076.

[44] Peters, R.A. and Strickland, R.N. (1990) Image complexity metrics for automatic target recognizers. *Automatic Target Recognizer System and Technology Conference* (Silver Spring, Maryland), pp. 30–31.

[45] Pierpont, J. (1959) *Lectures on the Theory of Functions of Real Variables*, Vol. 1 (Dover Publications, New York).

[46] Plataniotis, K.N. and Venetsanopoulos, A.N. (2000) *Color Image Processing and Applications* (Springer, Berlin).

[47] Pratt, W. (1978) *Digital Image Processing*, Vols. 1–2 (Wiley, New York).

[48] Roddier, F. (1981) The effects of atmospheric turbulence in optical astronomy. In Wolf, E. (ed.) *Progress in Optics*, Vol. 19 (Amsterdam, North-Holland), pp. 281–376.

[49] Rudin, L.I., Osher, S., and Fatemi, E. (1992) Nonlinear total variation based noise removal algorithms. *Physica D: Nonlinear Phenomena*, 60, 259–268.

[50] Sheikh, H.R., Bovik, A.C., and De Veciana, G. (2005) An information fidelity criterion for image quality assessment using natural scene statistics. *IEEE Transactions on Image Processing*, 14(12), 2117–2128.

[51] Sheikh, H.R., Sabir, M.F., and Bovik, F.C. (2006) A statistical evaluation of recent full reference image quality assessment algorithms. *IEEE Transactions on Image Processing*, 15(11), 3440–3451.

[52] Tikhonov, A.N. and Arsenin, V.Y. (1977) *Solutions of Ill-Posed Problems* (Wiley, New York); (Nauka, Moscow) 1979.

[53] Tomasi, C. and Manduchi, R. (1998) Bilateral filtering for gray and color images. *Proceedings of IEEE 6th International Conference on Computer Vision* (Bombay, India) pp. 839–846.

[54] Tomita, F. and Tsuji, S. (1977) Extraction of multiple regions by smoothing in selected neighborhoods. *IEEE Transactions SMC*, 7, 107–109.

[55] Tsukkerman, I.I., Katz, B.M., Lebedev, D.S., *et al.* (1981) *Digital Coding of Television Images* (Radio and Communication, Moscow) (in Russian).

[56] Tukey, J.W. (1971) *Exploratory Data Analysis* (Addison-Wesley, Reading).

[57] Varakin, L.E. (1970) *Theory of Complex Signals* (Sovetskoe Radio, Moscow) (in Russian).

[58] Vitushkin, A.G. (1959) *Estimation of the Complexity of the Tabulation Problem* (Fiz-Mat. Lit., Moscow) (in Russian).

[59] Vitushkin, A.G. (1955) *On Multidimentional Variations* (Gostekhizdat, Moscow) (in Russian).

[60] Wang, D., Vagnucci, A., and Li, C. (1981) Image enhancement by gradient inverse weighted smoothing scheme. *Computer Graphics Image Processing*, 15(2), 167–181.

[61] Wang, Z. and Bovik, A.C. (2006) *Modern Image Quality Assessment* (Morgan and Claypool, New York).

[62] Wang, Z., Bovik, A.C., Sheikh, H.R., and Simoncelli, E.P. (2004) Image quality assessment: From error visibility to structural similarity. *IEEE Transactions on Image Processing*, 13(4), 600–612.

[63] Wang, Z., Simoncelli, E.P., and Bovik, A.C. (2003) Multi-scale structural similarity for image quality assessment. *IEEE Asilomar Conference on Signals, Systems and Computers* (San Francisco, USA), pp. 1398–1402.

[64] Yu, H. and Winkler, S. (2013) Image complexity and spatial information, *Proceedings of 5th International Workshop on Quality of Multimedia Experience* (QoMEX) (Klagenfurt, Austria), pp. 12–17.

© 2025 World Scientific Publishing Company
https://doi.org/10.1142/9789811267215_0004

Chapter 4

Representation and Analysis of Regular Textures in Radon Space

Ivan G. Kazantsev

Institute of Computational Mathematics and Mathematical Geophysics of Siberian Branch of Russian Academy of Sciences, Novosibirsk, the Russian Federation
kig@ooi.sscc.ru

The Radon transform is a well-known technique in computer vision and image analysis, where it is used for detecting linear structures and curvilinear features in digital images, and is treated mostly as the Hough transform. The idea of directional structures consists of the back projection procedure, which is the main stage of almost all tomographic algorithms. The back projection, or a ridge function, follows the paths along which projections are gathered or numerically generated. We use the ridge function and its directions as the main features describing elongated structures and oriented textures.

Some of them such as linear parts of roads and buildings in an aerial image are a single ridge function. Other ones, such as structural textures in textiles, are supposed to be a sum of a few ridge functions. In computer vision applications, such as automated control of surfaces of materials, the detection of structural defects in textures is a specific problem of image analysis. Texture inspection as a distinct class of tasks includes, among others, inspections of textiles, ceramics, banknotes, hardwoods, metals, and papers. Many textures look like a superposition of elongated structures, therefore we analyze them within the framework of the

Radon space. We present an algorithm for the decomposition of a regular texture into informative ridge components. The results of numerical experiments are illustrated.

Keywords: Radon transform, reconstruction, ridge function, texture.

4.1 Introduction

The Radon transform is one of the main mathematical tools for solving tomography tasks [26]. The Radon transform is also a well-known technique in computer vision and image analysis, where it is used for detecting lines and curvilinear features in digital images and treated mostly as the Hough transform [25]. The Radon–Hough technique exploits the basic property of integrals along the line paths in accumulating values of an image along the structures of interest so that one can interpret a local peak in the projection space as a sign of the presence of a linear structure [17]. The idea of directional structures consists of the back-projection procedure, which is the main stage of almost all tomographic algorithms. The back projection, or a ridge function, follows the paths along which projections are gathered or numerically generated.

In this chapter, we use the ridge function [31] and its directions as the main features describing elongated structures and oriented textures. Some of them such as the linear part of roads and buildings in an aerial image are a single ridge function. Other ones, such as structural textures in textiles, are supposed to be a sum of a few ridge functions. In computer vision applications, such as automated control of surfaces of materials, the detection of structural defects in textures is a specific problem of image analysis [27].

The surfaces of many industrial materials are periodic structural textures, consisting of mostly elongated structures, superimposed in a regular but unknown way. Texture inspection as a distinct class of tasks includes, among others, inspections of textiles, ceramics, banknotes, hardwoods, metals, and papers. Many textures look like a superposition of elongated structures, therefore we analyze them within the framework of the Radon space.

In this chapter, we explore the ridge functions and their orientation as indicators of the detection of repetitive structures in image processing. An interesting problem is whether the decomposition of a texture into the sum of certain (most informative) ridge

functions would make sense for a low-level analysis of structural (non-stochastic) textures and what the numerical aspects of the task implementation are. In this chapter, effort to answer this question, mostly schematically, has been undertaken. To this end, we investigate the ridge functions within the framework of the Radon transform, or projection space, in the case of the two-dimensional parallel beam lines geometry.

We consider a ridge function as a model of a surface that is homogeneous in some direction and call it a single elongated structure. The detection of lines in an image exploits the observation that the projection data computed for this image includes the integrals along the lines that coincide with the desired line structures since these integrals accumulate the same (approximately) values along the structure. The presence of the line manifests itself in local extrema of the Radon transform. The same idea of the oriented structure is contained in the inverse projection operator and in the closely related concept of a ridge function. The recognition of a line assumes its dominant position in the image in terms of its amplitude and length.

Many texture analysis methods have been developed for visual inspection over the past few decades. The texture classification based on the co-occurrence matrix and combined feature descriptors is the subject of publication [1]. The rotation and scaling invariants inherently connected to the Radon transform are investigated in abundance of papers [4, 34, 35] for their use in texture classification. The Radon space-based feature descriptors, projection profiles, and recently developed ridgelets find their application for arranging the structural texture [17, 21, 32, 40]. Viewing the texture as a matrix, a subject of linear algebra, leads the authors to an interesting concept of low-rank textures [20]. The classical techniques in the field of image processing, such as the Gabor filters, the Laplacian filters, the wavelet decomposition, and the edge segmentation, yield useful results in texture filtering [14, 28]. The local binary patterns behave as the noise-robust approach in the texture image classification. [15, 19] The statistical and spectral characteristics of textures are important and widely used in both structural and stochastic texture exploration [7, 36].

Impressive results have been obtained with classical methods of partial differential equations (including the diffusion and transport equations) applied in the new field of "inpainting" [3], stochastic geometry and functional analysis [8], and methods of

optimization [9]. To fill in possible gaps in our review of the texture classification area, we should mention conventional methods, such as median filters [10, 22], higher-order statistics [13], and texture elements (textons) distribution [39].

We refer the reader to reviews and comparative studies of the texture recognition methods in [5, 6, 18, 25, 37]. Although books on texture analysis are rare, there have recently been published monographs and textbooks, which present a collection of high-quality research in the field [12,24,29,30]. We would like to cite [24]: "Despite all of the work done on texture, the definitions of texture that are used are rather vague. This illustrates the complexity of the texture phenomenon and science's lack of understanding of it." Starting with the fundamental paper by Robert Haralick [11], the contributions to unifying the structural and statistical pattern recognition are still to be resolved.

The remainder of this chapter proceeds as follows. In Section 4.2, we introduce the main methods relevant to image reconstruction using the ridge functions. In Section 4.3, we show the results of numerical experiments with the texture decomposition. In Section 4.4, we outline a scheme for detecting the local defects in a regular texture, based on the decomposition of a regular texture into informative ridge components. Section 4.5 finalizes this chapter.

4.2 Method

Definition 4.1. The Radon transform $\mathcal{R}$ (N-dimensional) maps function f, defined in $\mathbb{R}^N$, into the set of its integrals over hyperplanes in $\mathbb{R}^N$. If $\theta \in \mathbb{S}^{N-1}$ is a unit vector of projection direction and $s \in \mathbb{R}^1$, then

$$\mathcal{R}[f](\theta, s) \stackrel{\text{def}}{=} \int_{\langle u, \theta = s \rangle} f(u)du = \int_{\theta}^{\perp} f(s\theta + v)dv \qquad (4.1)$$

is the integral of f over hyperplane $\theta^{\perp}$ perpendicular to θ with distance $s \in \mathbb{R}^1$ to the origin, $\theta^{\perp} = \{u \in \mathbb{R}^N | \langle u, \theta \rangle = 0\}$.

In what follows, we consider the case $N = 2$. We denote operator $\mathcal{R}$ in this case as R [26]. The line through point u on the plane (x, y) can be represented in the form $\langle u, \theta \rangle = s$, where vector

$\theta = \theta(\alpha) = (\cos\alpha, \sin\alpha) \in \mathbb{S}^1$ — unit vector, scalar α is an angle between vector θ and positive axis y. Then

$$R_\theta[f](s) = \int_{\langle u,\theta\rangle = s} f(u)du \stackrel{\text{def}}{=} \int_{-\infty}^{\infty} f(s\theta + t\theta^\perp)dt. \tag{4.2}$$

Inverse Radon transform [26] using a full set of data $p(\theta, s) \equiv R[f](\theta, s)$ reads as

$$f = \left(\frac{1}{4\pi}\right) R^\# \mathcal{H}\left[\frac{\partial}{\partial s} p\right], \tag{4.3}$$

where $\mathcal{H}$ is the Hilbert transform with respect to the second variable

$$\mathcal{H}[p](\theta, s) \stackrel{\text{def}}{=} \int_{-\infty}^{\infty} \frac{p(\theta, t)}{s - t} dt \tag{4.4}$$

and $R^\#$ is the back-projection operator

$$R^\#[p](u) = \int_0^{2\pi} p(\theta, \langle u, \theta\rangle)d\theta. \tag{4.5}$$

Operator $R^\#$ averages over all lines crossing the point $u \in \mathbb{R}^2$. Given vector θ fixed, function $p(\theta, \langle u, \theta\rangle)$ is constant along lines, which are perpendicular to θ. These functions are known as ridge functions and equations (4.3)–(4.5) describe a decomposition of the f into the superposition of back-projections or ridge functions [31]. The Radon formula (4.3) lies in the foundation of the filtered back-projection (FBP) technique, where operator $(1/4\pi)\mathcal{H}(\partial/\partial \mathrm{s})$ is approximated by convolution with some kernel k, and the reconstruction algorithm reads as

$$f \;=\; R^\#[(k * p)], \tag{4.6}$$

$$(k * p)(\theta s) \stackrel{\text{def}}{=} \int_{-\infty}^{\infty} k(s - t)p(\theta, t)dt. \tag{4.7}$$

The formulas obtained use full data with respect to variables (θ, s) essentially. In practice, the data are discrete and the transition from the sampled values to full data continues to be an important task. The regularized representations of the kernel k and smooth data approximations (for example, based on splines), as well as other

theoretical and practical problems of solving ill-posed problems, are well studied in the literature and references therein [26, 33].

The investigation of the Radon transform often requires definiteness in the specification of the support of the function f. The important cases include the whole plane, convex bounded domain with smooth boundary, and polygons. We note that the inversion formulas and the formalism of their derivation essentially depend on the form of the support. For instance, there exists (to the best of our knowledge) only unique inversion formula [38] for a square support, which is of a number-theoretic nature and it differs from most of the formulas derived for the whole plane and circular disk-based Radon transforms. In what follows, we interpret functions f as images with support in the form of a circular disk.

Let f be a square-integrable function, which vanishes outside the unit disk D in the plane (x, y), i.e. $f \in L^2(D)$. The Radon transform maps function f onto its line integrals (4) along some direction vector θ determined by an angle $\alpha \in [0, \pi)$ and a distance $s \in [-1, 1]$ from the origin:

$$R_\alpha[f](s) = \int_{-a}^{a} f(s\cos\alpha - t\sin\alpha, s\sin\alpha + t\cos\alpha)dt, \qquad (4.8)$$

where $a = (1 - s^2)^{1/2}$ Map $f \longmapsto R_\alpha[f]$ is continuous from $L^2(D)$ into $L^2([-1, 1](1 - s^2)^{-1/2})$. We denote a single projection as $p(\alpha, s) = p_\alpha(s)$ and refer to it as a complete projection if it is known for all points in the interval $s \in [-1, 1]$. The inverse Radon transform is given by formula (4.3). This formula supposes the total knowledge of $p(\alpha, s), \alpha \in [0, \pi), s \in [-1, 1]$, thus an infinite number of complete projections. In practical tomographic reconstructions, only a limited number of projections and a finite number of samples per projection are available. If the set of directions of the projections is the n-tuple $\omega = (\omega_1, \ldots, \omega_n) \in [0, \pi)^n$, then we denote the corresponding set of complete projections by

$$R_\omega f = (R_{\omega_1} f, \ldots, R_{\omega_n} f). \qquad (4.9)$$

In tomographic scanners, recording a suficient number of projections with equally distributed direction angles provides the reconstruction of the function f from the data $R_\omega f$ in the form of an approximation

named filtered back-projection (FBP) that reads as

$$f(x,y) \cong \sum_{i=1}^{n} r_i(x\cos\omega_i + y\sin\omega_i). \tag{4.10}$$

The function r_i is defined as $r_i(s) = \int_{-\infty}^{\infty} p(\omega_i, t)k(s-t)dt$. The convolution kernel k is a regularized version of the inverse Fourier transform of the spatial frequency magnitude $|\xi|$. For each r_i, the same kernel k is used [26].

Formula (4.10) expresses the approximation of function f as a superposition of ridge functions r_i with directions ω_i uniformly distributed in the interval $[0,\pi)$. However, in the case of an arbitrary set of directions $\omega = (\omega_1,\ldots,\omega_n)$, different convolution kernels k–s have to be used for different directions. Fortunately, the alternative approach (Logan and Shepp) exists based on theorem [23], which states that if H is a function from $L^2(D)$ with the smallest norm-satisfying conditions

$$R_{\omega_i}[f] = R_{\omega_i}[H], \quad i = 1,\ldots,n, \tag{4.11}$$

then there exist functions $h_1,\ldots,h_n \in L^2([-1,1],(1-s^2)^{-1/2})$, such that

$$H(x,y) \equiv H_\omega(x,y) = \sum_{i=1}^{n} h_i(x\cos\omega_i + y\sin\omega_i). \tag{4.12}$$

Functions $h_i(x\cos\omega_i + y\sin\omega_i)$ are ridge functions. The Logan–Shepp theorem is an assertion of the existence of ridge functions in selected directions. We obtain a constructive algorithm for computing the superposition of ridge functions $H_\omega(x,y)$ from an arbitrary set of complete projections $R_\omega f$. It is known that for the minimal norm solution H_ω the identity is valid

$$\|f - H_\omega\|^2 = \|f\|^2 - \|H_\omega\|^2. \tag{4.13}$$

It follows from (4.13) that knowledge of the norm $\|H_\omega\|^2$ is useful in estimating the accuracy of reconstruction (4.12) for an arbitrary set of projections.

Let $H \in L^2(D)$ be defined as a sum (4.12) of n ridge functions. We denote $p_i(s) \equiv p(\omega_i, s)$ for convenience. Equations (4.11) form a

consistent system of equations for the reconstruction of function f. On the other hand, the system is a forward model for generating the projections p_i for the function H. Then we can formulate the problem of reconstructing the function H within the framework of the functions h_i and p_i of single variables. The suggested approach relies on the following propositions.

Proposition 4.1. *Let us assume* $\boldsymbol{H_\beta(x,y) = h_\beta(x\cos\beta + y\sin\beta), h_\beta(s) \in L^2([-1,1],(1-s^2)^{-1/2})}$ *is a ridge function for some angle* β*. Then its Radon transform or projection in arbitrary direction* α *reads as*

$$
\begin{aligned}
R_\beta[H_\beta](s) &= 2(1-s^2)^{1/2} h_\beta(s),\\
R_\alpha[H_\beta](s) &= (1/\sin(\alpha-\beta)) \int_{s1}^{s2} h_\beta(t)dt,\\
s_1 &= s\cos(\alpha-\beta) - (1-s^2)^{1/2}\sin(\alpha-\beta),\\
s_2 &= s\cos(\alpha-\beta) + (1-s^2)^{1/2}\sin(\alpha-\beta).
\end{aligned}
$$

Proof. We have

$$
\|H_\beta\|^2_{L^2(D)} = \left(\frac{1}{2}\right)\int_{-1}^{1}(1-s^2)^{(-1/2)} h_\beta^2(s)ds < \infty.
$$

Since $H_\beta \in L^2(D)$, we can apply the Radon transform R_α using (4.8)

$$
\begin{aligned}
R_\alpha[H_\beta](s) &= \int_{-\sqrt{1-s^2}}^{\sqrt{1-s^2}} H_\beta(s\cos\alpha - z\sin\alpha, s\sin\alpha + z\cos\alpha)dz\\
&= \int_{-\sqrt{1-s^2}}^{\sqrt{1-s^2}} h_\beta(s\cos(\alpha-\beta) - z\sin(\alpha-\beta))dz.
\end{aligned}
$$

Variables change $t = s\cos(\alpha-\beta) - z\sin(\alpha-\beta)$ proves the proposition. □

Proposition 4.2. *If function* $H(x,y)$ *is a sum of ridge functions, then the projection data constraints* $R_{\omega_i}[H] = p_i, (i = 1,\ldots,n)$ *are equivalent to the Fredholm system of integral equations*:

$$\mathrm{a(s)}h_i(s) + \sum_{j=1,j\neq i}^{n} \frac{1}{\sin(\omega_i - \omega_j)} \int_{s_1}^{s_2} h_j(t)dt = p_i(s), \quad (4.14)$$

$$\mathrm{a(s)} = 2(1-s^2)^{1/2}, \quad s \in [-1,1],$$
$$s_1 = s\cos(\omega_i - \omega_j) - (1-s^2)^{1/2}\sin(\omega_i - \omega_j),$$
$$s_2 = s\cos(\omega_i - \omega_j) + (1-s^2)^{1/2}\sin(\omega_i - \omega_j).$$

Proof. Introduce the ridge function $H_{\omega_j}(x,y) = h_j(x\cos\omega_j + y\sin\omega_j)$ Due to linearity of the Radon transform, the constraints become

$$R_{\omega_i}\left[\sum_{j=1}^{n} H_{\omega_j}\right](s) = R_{\omega_i}[H_{\omega_i}](s) + \sum_{j=1,j\neq i}^{n} R_{\omega_i}[H_{\omega_j}](s).$$

We obtain equation (4.14) using Proposition 4.1. □

Proposition 4.3. *The Chebyshev polynomials of the second kind*

$$U_{k-1}(t) = \sin(k \arccos t)/\sin(\arccos t), \quad k = 1,2,\ldots$$

satisfy the relationships

$$\int_{s_1}^{s_2} U_{k-1}(t)dt = a(s)\frac{\sin(k\alpha)}{k}U_{k-1}(s),$$
$$s_1 = s\cos(\alpha) - (1-s^2)^{1/2}\sin(\alpha),$$
$$s_2 = s\cos(\alpha) + (1-s^2)^{1/2}\sin(\alpha).$$

Proof. We use the identity $T_k^{'}(t) = kU_{k-1}(t)$, where $T_k(t) = \cos(k\arccos t)$ are the Chebyshev polynomials of the first kind. Substituting $\cos\beta = s, \sin\beta = (1-s^2)^{1/2}$, we obtain

$$\int_{\cos(\beta+\alpha)}^{\cos(\beta-\alpha)} \frac{T_k^{'}(t)}{k}dt = \frac{1}{k}[T_k(\cos(\beta-\alpha) - T_k(\cos(\beta+\alpha))]$$
$$= \frac{2}{k}\sin(k\arccos s)\sin(k\alpha) = \frac{a(s)}{k}U_{k-1}(s)\sin(k\alpha),$$

which proves the proposition. □

Proposition 4.4. *We are given with n-tuple* $\omega = (\omega_1, \dots, \omega_n) \in [0, \pi)^n$ *of distinct angles. Let* $f \in L^2(D)$ *and* $H \in L^2(D)$ *be the unique function of the smallest norm, which satisfy the equations*

$$R_{\omega_i}[f] = R_{\omega_i}[H], \quad i = 1, \dots, n. \tag{4.15}$$

Then there exist functions $h_1, \dots, h_n$ *such that*

$$H(x, y) = \sum_{i=1}^{n} h_i(x \cos \omega_i + y \sin \omega_i), \tag{4.16}$$

$$h_i(s) = \frac{1}{\pi} \sum_{k=1}^{\infty} \sum_{j=1}^{n} \eta_{ij}^{(k)} U_{k-1}(s) \int_{-1}^{1} p_j(t) U_{k-1}(t) dt, \tag{4.17}$$

where $\eta_{ij}^{(k)}$ *are entries of the matrix* Λ_k^+, *the generalized inverse of matrix* $\Lambda_k = \{\lambda_{ij}^{(k)}\}, i, j = 1, \dots, n$,

$$\lambda_{ij}^{(k)} = \sin(k(\omega_i - \omega_j))/(k(\omega_i - \omega_j)).$$

Proof. It can be easily seen that

$$h_i, p_i \in L^2\left([-1, 1], \left(1 - s^2\right)^{(-1/2)}\right) \subset L^2\left([-1, 1], \left(1 - s^2\right)^{(1/2)}\right).$$

Then denoting $v_i(s) \equiv p_i(s)/a(s)$, we have

$$h_i, v_i \in L^2([-1, 1](1 - s^2)^{(1/2)},$$

therefore we can apply the expansion of h_i, v_i into series with Chebyshev polynomials as the basis

$$\begin{aligned} h_i(s) &= \sum_{k=1}^{\infty} b_i^{(k)} U_{k-1}(s), \quad v_i(s) = \sum_{k=1}^{\infty} c_i^{(k)} U_{k-1}(s), \\ b_i^{(k)} &= \frac{2}{\pi} \int_{-1}^{1} (1 - t^2)^{1/2} h_i(t) U_{k-1}(t) dt, \\ c_i^{(k)} &= \frac{2}{\pi} \int_{-1}^{1} (1 - t^2)^{1/2} v_i(t) U_{k-1}(t) dt = \frac{1}{\pi} \int_{-1}^{1} p_i(t) U_{k-1}(t) dt. \end{aligned} \tag{4.18}$$

Substitute (4.18) into system (4.14) and using Proposition 3, we obtain

$$a(s)\sum_{k=1}^{\infty}\left(\sum_{j=1}^{n}\lambda_{ij}^{(k)}b_j^{(k)}-c_j^{(k)}\right)U_{k-1}(s)=0,\quad s\in[-1,1],\qquad(4.19)$$

where

$$\lambda_{ij}^{(k)}=\sin(k(\omega_i-\omega_j))/(k(\omega_i-\omega_j)).$$

Systems (4.14) and (4.19) are consistent and hence the system

$$\sum_{j=1}^{n}\lambda_{ij}^{(k)}b_j^{(k)}=c_i^{(k)},\quad i=1,\ldots,n,\qquad(4.20)$$

is consistent as well. Matrix $\Lambda_k=\{\lambda_{ij}^{(k)}\}$ can be singular (for instance, for $k=1$). In such case, the normal minimal solution of system (4.20) is calculated using the generalized inverse of matrix Λ_k. Finally, we represent the solution of (4.19) in the form

$$h_i(s)=\sum_{k=1}^{\infty}U_{k-1}(s)\sum_{j=1}^{n}\eta_{ij}^{(k)}c_j^{(k)},$$

with $\Lambda_k^{+}=\eta_{ij}^{(k)}$. As a result, we obtain equation (4.17) or, in other terms, the singular value decomposition (SVD) solution [26]. □

Corollary 4.1 *The norm of H can be calculated using projections $p_i(s)$*

$$\|H\|^2=\frac{1}{\pi}\sum_{k=1}^{\infty}\sum_{j=1}^{n}\eta_{ij}^{(k)}\int_{-1}^{1}U_{k-1}(s)p_i(s)ds\int_{-1}^{1}U_{k-1}(t)p_i(t)dt.$$

Proof. The result follows from Proposition 4.4 and identity

$$\|H\|^2=\sum_{i=1}^{n}\int_{-1}^{1}h_i(s)p_i(s)ds.$$

□

If the angles ω_i are equally spaced $\omega_i = \frac{\pi}{n}(i-1), i = \overline{1,n}$), then the analytical inversion of the matrix Λ_k is possible. In the case of equally spaced angles, we can reduce the calculation to a computationally more eficient formula [16]:

Proposition 4.5. *If $k \leq n$, matrix Λ_k is singular and its generalized inverse Λ_k^+ is*

$$\Lambda_k^+ = \left(\frac{k^2}{n^2}\right)\Lambda_k. \tag{4.21}$$

Let us assume that the image f is known explicitly, as is the case when dealing with tomographic phantoms in numerical experiments or digital images in computer vision tasks. We consider n projections

$$R_\omega f = (p_{\omega_1}, \ldots, p_{\omega_n})$$

of function f as its representatives or features. Varying the n-tuple ω, we can calculate the best approximation of the function f by a sum of corresponding ridge functions

$$H_\omega = \sum_{i=1}^{n} h_{\omega_i}.$$

An interesting property of this representation is that the minimal solution is unique but not the set of ridge functions generating the solution. The projection data $R_\omega f$ obtained after global search maximally stores information about the elongated structures present in the image.

Definition 4.3. We shall refer to the quantity $Q(f,\omega) \equiv \|H_\omega\|^2$ as information content or the informativity of the projection set $R_\omega f$.

Hence, we say that the projection set $R_\omega f$ is more informative than the set $R_\alpha f$ if inequality $Q(f,\omega) > Q(f,\alpha)$ holds. We estimate the information content of projection data $R_\omega f$ by the norm of the minimal norm solution H_ω due to relationship (4.13). Given an image f and a number n, we can find the most informative tuple ω_0 by a global search

$$\omega_0 = \mathrm{argmax}_\omega Q(f,\omega). \tag{4.22}$$

We can show that the information content of a single projection p_{ω_j} is

$$Q(f,\omega_j) = \frac{1}{2}\int_{-1}^{1}(1-s^2)^{-1/2}p_{\omega_j}^2. \tag{4.23}$$

4.3 Numerical Experiments with Texture Decomposition

In our numerical experiments, we extensively use the texture D52 "Oriental ornament" shown in Figure 4.1 (left) from the album of textures by Brodatz [2]. The texture image is of 128×128 pixels size. We have numerically generated 180 projections within the range $[0, \pi)$ and with 128 samples per projection (Figure 4.1, right).

We have calculated the values of the information content of individual projections using formula (4.24) and presented these values in Figure 4.2 (left). For the D52 texture, four informative directions turned out to be $\omega = (0°, 48°, 90°, 131°)$. The corresponding normal solution is shown in Figure 4.2 (right).

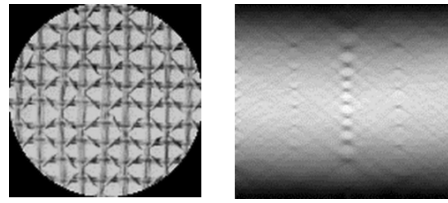


Figure 4.1. (Left) Test image f with squared norm $\|f\|^2 = 85473$. (Right) The Radon projections of D52 texture occupy columns of the Radon space image with directions $0°, 1°, \ldots, 179°$.

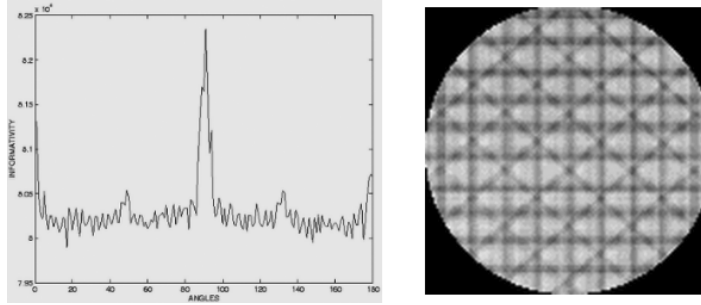


Figure 4.2. (Left) Values of the information content Q for single projections with four local maxima $\omega = (0°, 48°, 90°, 131°)$. (Right) Reconstruction H_ω from four informative projections with the directions ω, $\|H_\omega\|^2 = 83757$, $\|H_\omega\|^2/\|f\|^2 = 0.98$.

4.4 Numerical Extraction of Texture from Background

Consider an additive model of image distortion in the form of the equation

$$g(x,y) = f(x,y) + b(x,y), \tag{4.24}$$

where g is the observed image, f is the desired useful image, and b is a certain interfering structure, which is known to be well approximated by the sum of a small number of the ridge functions in n directions $\omega = (\omega_1, \ldots, \omega_n)$. The task is to restore f from the data of g. Due to additivity of model (4.24) and linearity of the Radon transform, the expansion of the functions g, f, b into the superposition of ridge functions in the directions ω results in the equality

$$G_\omega = F_\omega + B_\omega, \tag{4.25}$$

where $G_\omega, F_\omega, B_\omega$ are the minimal normal solutions obtained, respectively, by reconstructing the images g, f, b from the datasets

$$R_\omega[g], R_\omega[f], R_\omega[b].$$

Since it is known that the texture b is well approximated by the ridge functions in the directions ω, we have $b \cong B_\omega$. Then subtracting equation (4.24) from (4.25), we approximately obtain

$$g(x,y) - G_\omega(x,y) \cong f(x,y) - F_\omega(x,y). \tag{4.26}$$

The numerical experiments with the test image "Cameraman" (Figure 4.3) show reconstructions with visually distinguishable details and much smaller artifacts as compared to the original distorted images.

The idea of the method is that since n is small, then the SVD-based decomposition F_ω is a very smooth blurred version of the function f. Therefore, the difference $f(x,y) - F_\omega(x,y)$ on the right-hand side of (4.26) is approximately similar to f with a shifted average value. Converting the image $g(x,y) - G_\omega(x,y)$ to the range of values of a visualizing device, we synthesize an

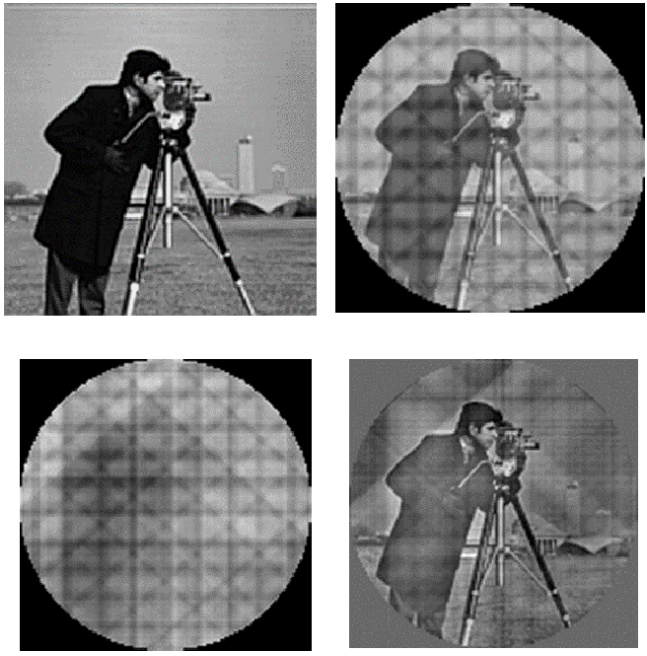


Figure 4.3. (Top left) The test image f "Cameraman". (Top right) The image $g = f + b$ is the sum of the test image f and image b of texture D52. (Bottom left) Ridge approximation G_ω of function g from four projections in directions $\omega = (0°, 48°, 90°, 131°)$, informative for the D52. (Bottom right) The image of the difference $g - G_\omega \cong f$.

image essentially free from distortions caused by the presence of the texture b.

We have performed computational experiments to illustrate the efficiency of the ridge functions as applied to the method of "inpainting" (interpolating) the damaged parts of an image and repair its surrounding areas. We consider the task of inpainting in terms of the Radon integral transform, where the ridge functions allow the use of anisotropy information in the analysis of regular textures in explicit form (Figure 4.4). The method used has the advantage that it is not necessary to know the location of corrupted regions, as required in the formulation of the method of "inpainting" in terms of partial differential equations.



Figure 4.4. (Left) The sum of cameraman image f and several horizontal and vertical white lines representing the background b. (Right) The result of the removal of the distorting lines by deleting the horizontal and vertical ridge components from the SVD-based representation of the sum of f and the background b using only the information of the lines' directions.

4.5 Conclusions

The results presented demonstrate that the methodology of the ridge functions is applicable to the problems of computer vision offering some hope for success. The benefits in the accuracy of decomposition into a few ridge functions remain somewhat problematic, although we do see images of quality superior to those produced by the filtered back-projection. In this context, the ridge functions constitute the background for embedding the directionality into the texture analysis. The texture reconstruction formulae used in this study benefit over the circular shape of the image support; we do not know how the formulae should be modified in the case of the square or more complex convex shape of an image. We conclude that the ridge function-based inversion formulae synthesize both analytical and algebraic features of the Radon transform.

The experimental results in the visual inspection of structural textures show certain progress in the decomposition of a texture into a sum of a few ridge functions, although the comparison with other texture operators was not done in order to prove superiority (or inferiority) of the texture approximation presented. The author considers these topics to be the subject worth future research.

Acknowledgments

This work was partially supported by the state contract with the Institute of Computational Mathematics and Mathematical Geophysics (project no. 0251-2021-0003).

References

[1] Barley, A. and Town, C. (2014) Combinations of feature descriptors for texture image classification. *Journal of Data Analysis and Information Processing*, 2, 67–76.

[2] Brodatz, P. (1966) Textures: A Photographic Album for Artists and Designers (Dover Publications, New York).

[3] Casaca, W., Boaventura, M., de Almeida, M.P., and Nonato, L.G. (2014) Combining anisotropic diffusion, transport equation and texture synthesis for inpainting textured images. *Pattern Recognition Letters*, 36, 36–45.

[4] Cui, P., Li, J., Pan, Q., and Zhang, H. (2006) Rotation and scaling invariant texture classification based on Radon transform and multiscale analysis. *Pattern Recognition Letters*, 27, 408–413.

[5] Dixit, A. and Hegde, N.P. (2013) Image texture analysis techniques — survey. Proceedings of 3rd International Conference on Advanced Computing & Communication Technologies, pp. 69–76.

[6] Eskenazi, S., Gomez-Krämer, P., and Ogier, J.-M. (2017) A comprehensive survey of mostly textual document segmentation algorithms since 2008. *Pattern Recognition*, 64, 1–14.

[7] Fedoseev, V.A. (2015) An algorithm for evaluating the spectral characteristics of regular textures. *Pattern Recognition and Image Analysis*, 25, 22–26.

[8] Fedotov, N.G. and Mokshanina, D.A. (2010) Recognition of halftone textures from the standpoint of stochastic geometry and functional analysis. *Pattern Recognition and Image Analysis*, 20, 551–556.

[9] Gilboa, G. (2014) A total variation spectral framework for scale and texture analysis. *SIAM Journal on Imaging Science*, 7, 1937–1961.

[10] Hafiane, A., Palaniappan, K., and Seetharaman, G. (2015) Joint adaptive median binary patterns for texture classification. *Pattern Recognition*, 48, 2609–2620.

[11] Haralick, R.M. (1979) Statistical and structural approaches to texture. *Proceedings of the IEEE*, 67, 786–804.
[12] Herout, A., Dubska, M., and Havel, J. (2013) *Real-Time Detection of Lines and Grids by PClines and Other Approaches*. Springer Briefs in Computer Science.
[13] Huang, Y. and Chan, K. (2015) Texture decomposition by harmonics extraction from higher order statistics. *IEEE Transactions on Image Processing*, 13, 1–14.
[14] Hui, D., Xiaogang, J., and Willis, P.J. (2015) Two-level joint local Laplacian texture filtering. *The Visual Computer*, 32, 1–12.
[15] Ji, Z. and Nie, L. (2016) Texture image classification with noise-tolerant local binary patterns. *Computer Research and Development*, 53, 1128–1134.
[16] Kazantsev, I.G. (1998) Tomographic reconstruction from arbitrary directions using ridge functions. *Inverse Problems*, 14, 635–645.
[17] Kim, H.-B. and Park, R.-H. (1992) Extracting spatial arrangement of structural textures using projection information. *Pattern Recognition*, 25, 237–245.
[18] Koltsov, P.P. (2011) Comparative study of texture detection and classification algorithms. *Computational Mathematics and Mathematical Physics*, 51, 1460–1466.
[19] Kylberg, G. and Sintorn, L.M. (2013) Evaluation of noise robustness for local binary patterns descriptors in texture classification. *EURASIP Journal on Image and Video Processing*, 13, 1–20.
[20] Liang, X., Ren, X., Zhang, Z., and Ma, Y. (2016) Texture repairing by unified low rank optimization. *Journal of Computer Science and Technology*, 31, 525–546.
[21] Liu, G., Lin, Z., and Yu, Y. (2009) Radon representation-based feature descriptor for texture classification. *IEEE Transactions on Image Processing*, 18, 921–928.
[22] Liu, L. and Fieguth, P. (2012) Texture classification from random features. *IEEE Transactions of Pattern Analysis and Machine Intelligence*, 43, 574–586.
[23] Logan, B.F. and Shepp, L.A. (1975) Optimal reconstruction of a function from its projections. *Duke Mathematical Journal*, 42, 661–706.
[24] Mirmehdi, M., Xie, X., and Suri, J. (2008) *Handbook of Texture Analysis* (Imperial College Press, London).
[25] Mukhopadhyaya, P. and Chaudhuria, B.B. (2015) A survey of Hough transform. *Pattern Recognition*, 48, 993–1010.
[26] Natterer, F. (1986) *The Mathematics of Computerized Tomography* (Teubner and Wiley, New York).

[27] Ngan, H.Y., Pang, G.K., and Yung, N.H. (2008) Motif-based defect detection for patterned fabric. *Pattern Recognition*, 41, 1878–1894.

[28] Pang, W., Choi, K., and Qin, J. (2016) Fast Gabor texture feature extraction with separable filters using GPU. *Journal of Real Time Image Processing*, 12, 5–13.

[29] Petrou, M. and Sevilla, P.G. (2006) *Image Processing. Texture: Dealing with Texture* (Wiley John and Sons, New York).

[30] Pietikainen, M.K. (2000) *Texture Analysis in Machine Vision* (World Scientific Publishing Co. Pte. Ltd., Singapore, 2000).

[31] Pinkus, A. (2015) *Ridge Functions.* Cambridge Tracts in Mathematics (Cambridge University, Cambridge).

[32] Qiao, Y.L., Song, C.Y., and Zhao, C.H. (2010) M-band ridgelet transform based texture classification. *Pattern Recognition Letters*, 31, 244–249.

[33] Ramlau, R. and Scherzer, O. (2018) The first 100 years of the Radon transform. *Inverse Problems*, 34, 090201.

[34] Ren, X. and Lin, Z. (2013) Linearized alternating direction method with adaptive penalty and warm starts for fast solving transform invariant low-rank textures. *International Journal of Computer Vision*, 104, 1–14.

[35] Shrivastava, N. and Tyagi, V. (2016) Noise-invariant structure pattern for image texture classification and retrieval. *Multimedia Tools and Applications*, 75, 10887–10906.

[36] Susan, S. and Hanmandlu, M. (2013) A non-extensive entropy feature and its application to texture classification. *Neurocomputing*, 120, 214–225.

[37] Uhl, A. and Wimmer, G. (2015) A systematic evaluation of the scale invariance of texture recognition methods. *Pattern Analysis and Applications*, 8, 945–969.

[38] Vvedenskaya, N.D. and Gindikin, S. (1984) Poisson formula for Radon transform and numerical algorithm for image reconstruction. *Doklady Mathematics*, 279, 780–784 (in Russian).

[39] Xie, J., Zhang L., You, J., and Shiu S. (2015) Effective texture classification by texton encoding induced statistical features. *Pattern Recognition*, 48, 447–457.

[40] Yalniz, I.Z. and Aksoy, S. (2010) Unsupervised detection and localization of structural textures using projection profiles. *Pattern Recognition*, 43, 3324–3337.

This page intentionally left blank

© 2025 World Scientific Publishing Company
https://doi.org/10.1142/9789811267215_0005

Chapter 5

Mathematical Model and Methods of Analysis of Continuous Contours Images

Rinat G. Khafizov[*,‡], **Sergei A. Okhotnikov**[*,§], **and Mikhail A. Egoshin**[†,¶]

[*]*Federal State Budgetary Educational Institution of Higher Education, Volga State Technological University, Yoshkar-Ola, the Russian Federation*

[†]*Federal State Budgetary Educational Institution of Higher Education, Nizhny Novgorod State Technical University n.a. R.E. Alekseev, Nizhny Novgorod, the Russian Federation*

[‡]*hafizovrg@volgatech.net*

[§]*ohotnikovsa@volgatech.net*

[¶]*egmforwrk@gmail.com*

The questions of representation, processing, and analysis of continuous deterministic and random image contours and estimation of their parameters are considered. An approach is proposed to describe continuous complex-valued signals represented on the complex plane in the form of closed contours. A linear space of vector contours is defined and the main analytical relations are obtained.

A model of a random continuous contour is proposed, which is a complex random function. In this case, the complex random function is considered as a set of its possible realizations. The concepts of mathematical expectation and variance of a random contour are introduced. Geometrically, the mathematical expectation of a random contour is interpreted as an "average contour" around which other contours

are located: realizations. The dispersion characterizes the degree of scattering of possible realizations (contours) around the mathematical expectation of a random contour (the "middle contour"). It is shown that an important condition for the formation of an adequate contour model with a random form is the equality of the values of the parameters of the linear transformations of the contours of its realizations. Alignment of these parameters should be performed during the formation of a contour model with a random form.

The problems of spectral and correlation analysis of continuous contours are considered and features of their spectra are revealed. The problems of discretization of continuous contours of images are investigated. The structure of the device for processing continuous contours of images and the results of its modeling are presented.

Keywords: Random contour, mathematical expectation, variance, continuous elementary contours, autocorrelation function, cross-correlation function, spectrum of a stationary random contour.

5.1 Introduction

The object's contours can often be considered as the most informative part of the image [1–4, 6]. The main issues of image contour analysis have been reviewed in the monograph "Introduction to contour analysis and its applications to image processing and signals", published under the editorship of Professor Ya. A. Furman in "FIZMATLIT" publishing house in 2002. In [1] are discusses approaches to image processing based on the analysis of image contours, which contains information about the shape of the object, its scale, and angular position. The contours of the images fully characterize their shape and provide a simple analytical description, which is invariant to shift, rotation, and scaling of images. Consideration of the image contours as complex-valued signals and their representation in the complex-valued linear space allows us to get the proximity of two contours as a scalar product that is invariant to transformations of shift, rotation, and scaling.

The number of pixels in the image has a significant impact on the ability to apply methods of contour analysis. A decrease in the number of pixels due to sampling noise decreases the signal/noise ratio. The masking effect of sampling noise first leads to the loss of the particular, small details of the images, i.e. to the loss of informative features in their contours [1]. The required number of pixels is associated with the contour spectral width and should be selected based on the

Nyquist theorem. However, in many cases, the number of pixels is determined by the capabilities of the programmable image processor and available image sensors.

In [1] it is emphasized that the question of application of the complex variable theory as a basis for processing and analysis of image contours is of particular interest. The theory of analytic functions of a complex variable provides many useful mathematical models. Mathematical models are simplified if we consider real variables as a special case of complex variables. Therefore, this monograph focuses on the analysis and processing of deterministic and random continuous image contours.

5.2 A Mathematical Model of Continuous Contour

5.2.1 *A continuous contour of deterministic form*

If for any natural number l a complex number $\chi(l) = \mathrm{Re}\chi(l) + i\mathrm{Im}\chi(l)$ is specified, then in this case we are dealing with a complex continuous variable defined on the plane C of a complex continuous variable χ. Representation of image contours as functions of complex variables allows using the theory of functions of a complex variable for their analysis and processing.

The contour $\mathbf{X} = \{\chi(l)\}_{0,L}$ as a continuous closed curve defined on the complex plane is represented as [7]

$$\chi(l) = \mathrm{Re}\chi(l) + i\mathrm{Im}\chi(l) = |\chi(l)| \exp\{i\varphi(l)\},$$

where l is any in the range from 0 to L, L is the contour length, $\mathrm{Re}\chi(l)$ and $\mathrm{Im}\chi(l)$ are real and imaginary components of the function $\chi(l)$, and $|\chi(l)| = \sqrt{\mathrm{Re}\chi(l)^2 + \mathrm{Im}\chi(l)^2}$ and $\varphi(l) = \arg\, \chi(l)$ are the amplitude and phase representation of the function $\chi(l)$. Figure 5.1 shows examples of continuous contours.

With multiple traversals along a closed line, the contour $\mathbf{X}$ can be represented as a periodic function with a period L, i.e.

$$\chi(l) = \chi(l + \alpha L), \quad \alpha = 0, 1, 2, \ldots. \tag{5.1}$$

The contours of the images will be considered as vectors in a linear complex space. $\chi(l)$ will be the current contour component and l is any in the range from 0 to L.

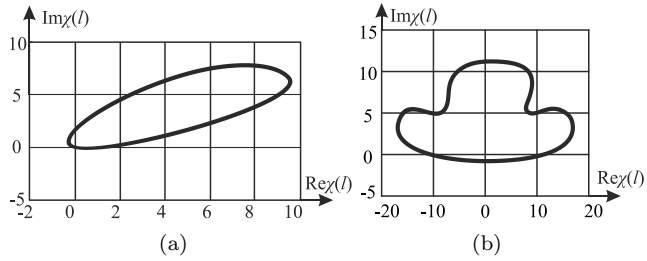


Figure 5.1. Examples of continuous contours.

Two contours $\mathbf{X}=\{\chi(l)\}_{0,L}$ and $\mathbf{Y}=\{\upsilon(l)\}_{0,L}$ are equal if $\chi(l)=\upsilon(l)$. Linear operations are defined by the relation

$$\mathbf{Y}=\mu\mathbf{X}=\{\mu\chi(l)\}_{0,\,L},$$

where $\mu=|\mu|\exp\{i\Delta\varphi\}$ is an arbitrary complex coefficient. When the contour $\mathbf{X}$ is multiplied by a complex factor μ, the contour is stretched by a $|\mu|$ factor and rotated by an angle $\Delta\varphi=\arg\mu$. This process can be described as a special case of a fractional-linear mapping of the form

$$\nu=\frac{a\gamma+b}{c\gamma+d},\quad (bc-ad)\neq 0.$$

This fractional-linear mapping establishes a one-to-one correspondence between the points of the γ-plane and the points of the ν-plane. The mapping $\nu=a\gamma+b$ where $a=a_1+ia_2$ and $b=b_1+ib_2$ are arbitrary complex numbers corresponds to rotation by angle $\arg a$, stretching by a $|a|$ factor, and parallel shift by a vector b.

The norm of a continuous contour is

$$\|\mathbf{X}\|=\sqrt{\int_0^L\chi(l)\chi^*(l)dl}=\sqrt{\int_0^L(\mathrm{Re}\chi(l)^2+\mathrm{Im}\chi(l)^2)dl}$$
$$=\sqrt{\int_0^L|\chi(l)|dl}. \tag{5.2}$$

The expression for the squared distance between $\mathbf{X}$ and $\mathbf{Y}$ contours is given by

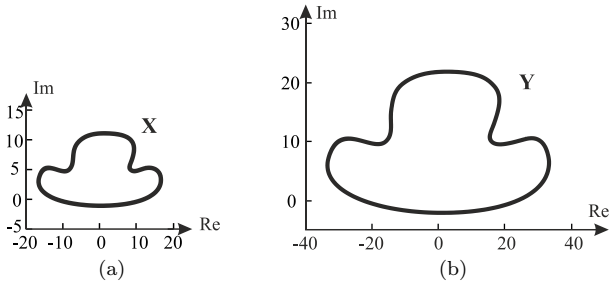


Figure 5.2. Continuous contour $\mathbf{X}$ (a) and its scale copy $\mathbf{Y}$ (b) with $\mu = 2$.

$$R^2 = (\mathbf{X}-\mathbf{Y},\ \mathbf{X}-\mathbf{Y}) = \int_0^L (\chi(l)-\upsilon(l))(\chi^*(l)-\upsilon^*(l))dl$$
$$= \int_0^L \chi(l)\chi^*(l)dl + \int_0^L \upsilon(l)\upsilon^*(l)dl - \int_0^L \chi(l)\upsilon^*(l)dl$$
$$- \int_0^L \upsilon(l)\chi^*(l)dl = \|\mathbf{X}\|^2 + \|\mathbf{Y}\|^2 - (\mathbf{X},\mathbf{Y}) - (\mathbf{Y},\mathbf{X}).$$

The scalar product of two continuous contours has the form

$$(\mathbf{X},\mathbf{Y}) = \int_0^L \chi(l)\upsilon^*(l)dl$$
$$= \int_0^L (\mathrm{Re}\,\chi(l) + i\mathrm{Im}\,\chi(l))(\mathrm{Re}\,\upsilon(l) - i\mathrm{Im}\,\upsilon(l))dl$$
$$= \int_0^L (\mathrm{Re}\,\chi(l)\mathrm{Re}\,\upsilon(l) + \mathrm{Im}\,\chi(l)\mathrm{Im}\,\upsilon(l))dl$$
$$+ i\int_0^L (\mathrm{Im}\,\chi(l)\mathrm{Re}\,\upsilon(l) - \mathrm{Re}\,\chi(l)\mathrm{Im}\,\upsilon(l))dl. \qquad (5.3)$$

Consider, for example, how a linear transformation of a continuous contour $\mathbf{X}$ impacts the distance and scalar product.

1. Scaling of a continuous contour
Let the contour $\mathbf{Y}$ be formed from the contour $\mathbf{X}$ represented in Figure 5.1(b), by changing the scale by a factor $\mu = 2$ (Figure 5.2).

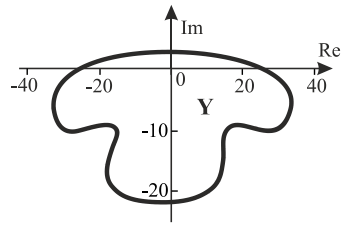


Figure 5.3. Continuous contour $\mathbf{Y}$, obtained by changing scale by a factor $\mu = 2$ and rotating it by an angle $\varphi = \pi$.

In accordance with expression (5.2), we find the norms of the initial continuous contour $\mathbf{X}$ and its large-scale copy $\mathbf{Y}$:

$$\|\mathbf{X}\| = 31{,}118;\ \|\mathbf{Y}\| = 62{,}236.$$

The scalar product of continuous contours $\mathbf{X}$ and $\mathbf{Y}$:

$$(\mathbf{X}, \mathbf{Y}) = \int_0^L \chi(l)\upsilon^*(l)dl = 1{,}937 \cdot 10^3,$$

and the squared of the distance between them is

$$R^2 = \|\mathbf{X}\|^2 + \|\mathbf{Y}\|^2 - 2(\mathbf{X}, \mathbf{Y}) = 968{,}332.$$

The normalized scalar product

$$\eta = \frac{(\mathbf{X}, \mathbf{Y})}{\|\mathbf{X}\|\,\|\mathbf{Y}\|} = \frac{1,937 \cdot 10^3}{31,118 \cdot 62,236} = 1.$$

Thus, the measure of similarity in the form of a normalized scalar product is invariant to a change in the scale of the analyzed contour.

2. Scaling and rotating a continuous contour

Let the contour $\mathbf{Y}$ be formed from the contour $\mathbf{X}$ represented in Figure 5.1(b), by changing the scale by a factor $\mu = 2$ and rotating by an angle $\varphi = \pi$ (Figure 5.3).

Contour $\mathbf{Y}$ norm

$$\|\mathbf{Y}\| = 62,236.$$

The scalar product of continuous contours $\mathbf{X}$ and $\mathbf{Y}$

$$(\mathbf{X}, \mathbf{Y}) = \int_0^L \chi(l)\upsilon^*(l)dl = -1,937 \cdot 10^3,$$

Distance is

$$R^2 = \|\mathbf{X}\|^2 + \|\mathbf{Y}\|^2 - 2(\mathbf{X}, \mathbf{Y}) = 8,517 \cdot 10^3.$$

The normalized scalar product

$$\eta = \frac{(\mathbf{X}, \mathbf{Y})}{\|\mathbf{X}\| \, \|\mathbf{Y}\|} = \frac{-1,937 \cdot 10^3}{31,118 \cdot 62,236} = -1.$$

The magnitude of η is 1, and the argument is π. Consequently, both scale and rotation transformation do not have any effect on the scalar product of the two continuous contours.

5.2.2 *Continuous contour with random form*

The shape of a significant number of natural objects surrounding us is random. Examples of random images can be images of plant leaves, human organs, clouds, the profile of a person's face, etc. (Figure 5.4). The essential variability of the form of such images does not allow us to consider them as noisy images of some reference objects. Nevertheless, a person confidently distinguishes them from each other and successfully recognizes such objects [1].

Let $\mathrm{X}(l) = \mathrm{Re}\ \mathrm{X}(l) + \mathrm{Im}\ \mathrm{X}(l)$ be a complex random function, i.e. function of a nonrandom argument l, which is a complex random variable for each fixed value of the argument. Here $\mathrm{Re}\ \mathrm{X}(l)$ and $\mathrm{Im}\ \mathrm{X}(l)$ are real random functions of the real argument l. We shall consider a complex random function $\mathrm{X}\ (l)$ as a set of its possible realizations $\chi_1(l), \chi_2(l), \ldots, \chi_n(l)$ and call it a random contour [10, 11].

The realization of a random contour $\mathrm{X}(l)$ is called a non-random function of the argument l, which can be equal to a random contour as a result of the test [5].

For a fixed value of the argument, for example, for $l = l_1$, we obtain a section: a random variable $\mathrm{X}(l_1)$ with mathematical expectation $M[\mathrm{X}(l_1)]$ and variance $\mathrm{D}[\mathrm{X}(l_1)] \geq 0$. Thus, each fixed value of the argument l defines a section: a random variable, and to each random variable, there corresponds a certain mathematical expectation and variance.

The mathematical expectation $m_{\mathrm{X}}(l)$ of a random contour $\mathrm{X}(l)$ is a non-random complex function whose value for each fixed value of the argument l is equal to the mathematical expectation of the

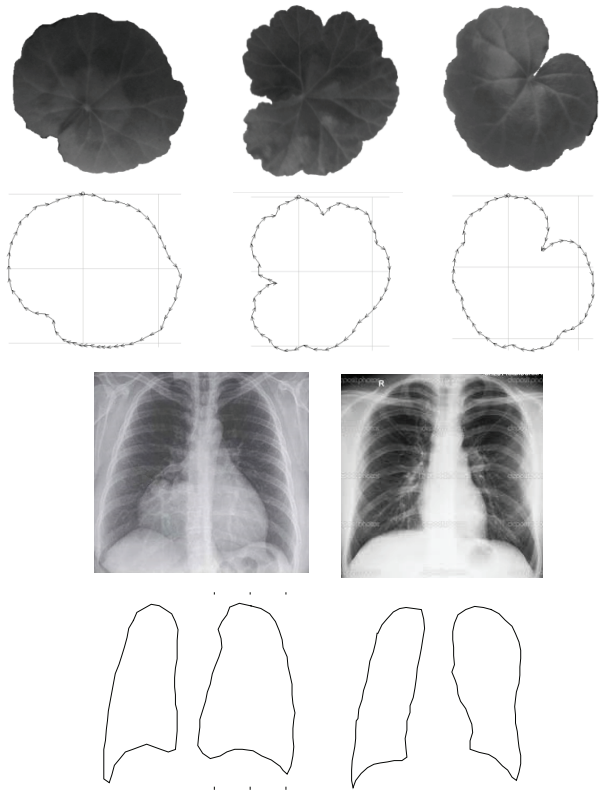


Figure 5.4. Examples of random shape images and their contours.

section corresponding to the same fixed value of the argument: $m_{\mathrm{X}}(l) = M[\mathrm{X}(l)]$. Wherein

$$m_{\mathrm{X}}(l) = m_{\mathrm{Re\,X}}(l) + i\, m_{\mathrm{Im\,X}}(l).$$

In a particular case, the function $m_{\mathrm{X}}(l)$ can retain a constant value for all valid values of the argument.

Geometrically, the mathematical expectation of a random contour can be interpreted as an "average contour", near which implementations, the other contours, are located (Figure 5.5).

For a fixed value of the argument, the mathematical expectation is the mean value of the section around which its possible values are located (Figure 5.6).

The variance $D_X(l)$ of a random contour $\mathrm{X}(l)$ is a non-random non-negative function whose value for each fixed value of the

argument l is equal to the variance of the section corresponding to the same fixed value of the argument: $D_X(l) = D[\mathrm{X}(l)]$. Wherein

$$D_{\mathrm{X}}(l) = M\left[|\overset{\circ}{\mathrm{X}}(l)^2|\right],$$

where $\overset{\circ}{\mathrm{X}}(l) = \mathrm{X}(l) - m_{\mathrm{X}}(l)$ is a centered random contour. Considering that the mathematical expectation of a sum is equal to the sum of the mathematical expectations of the terms, we get

$$\begin{aligned} D_{\mathrm{X}}(l) &= M\left[|\overset{\circ}{\mathrm{X}}(l)^2|\right] = M\left[\mathrm{Re}\,\overset{\circ}{\mathrm{X}}(l)^2 + \mathrm{Im}\,\overset{\circ}{\mathrm{X}}(l)^2\right] \\ &= M\left[\mathrm{Re}\,\overset{\circ}{\mathrm{X}}(l)\right]^2 + M\left[\mathrm{Im}\,\overset{\circ}{\mathrm{X}}(l)\right]^2 = D_{\mathrm{ReX}} + D_{\mathrm{ImX}}. \end{aligned}$$

Thus, the variance of a complex random function is equal to the sum of the variances of its real and imaginary parts. In a particular case $D_X(l)$, it can retain a constant value for all valid values of the argument.

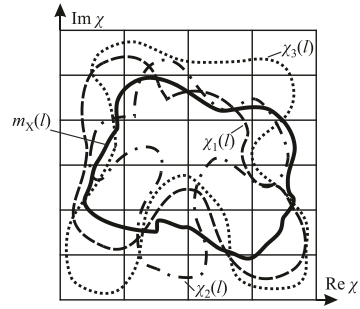


Figure 5.5. Geometric interpretation of the mathematical expectation $m_{\mathrm{X}}(l)$ of a random contour $\mathrm{X}(l)$.

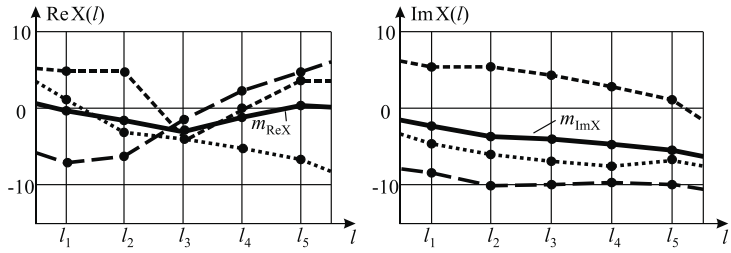


Figure 5.6. The mathematical expectation of sections of a random contour $\mathrm{X}(l)$ for fixed values of the argument.

The dispersion characterizes the variance degree of possible realizations around the mathematical expectation of a random contour (the "average contour"). For a fixed argument, the variance characterizes the dispersion degree of possible values of the cross-section around the mathematical expectation of the cross-section.

An important condition for the formation of an adequate model of random shape contour is the equality of the parameters of the linear transformations of its realizations. They should have the same scale factor $|\mu|$ and zero value of the angle $\Delta\varphi$ of mutual rotation, and the starting points a_0 must coincide. Alignment of these parameters should be done in the process of forming a random shape contour model.

Let us consider an example of the formation of a mathematical model of a random contour. Let there be 10 images of the rose leaves (Figure 5.7).

Selecting the outlines, we get 10 realizations of a random contour of the type "Rose". Next, we correct the rotation angle, scale, and initial point shift of the realizations of the random contour of the type "Rose".

Figures 5.8 and 5.9 show the implementation of a random contour of the type "Rose" after parameters' correction.

In Figure 5.10, (a) represents the mathematical expectation contour obtained from the implementations in Figures 5.8, 5.9, and Figure 5.10(b), variance of a random contour of the "Rose" type.

On average, the dispersion of a random contour of the type "Rose", represented by 10 realizations (see Figures 5.8 and 5.9), is equal to $1{,}047 \cdot 10^{-3}$.

5.3 Correlation and Spectral Analysis of Continuous Contours

5.3.1 *Spectral analysis of continuous contours*

5.3.1.1 *Orthogonal basis*

Let us set an orthogonal basis on which the random continuous contour decomposes:

$$\mathbf{X}_m = \{\chi_m(l)\}_{0,L} = \left\{ |\chi| \exp\left\{ i\frac{2\pi}{L} ml \right\} \right\}_{0,L}. \tag{5.4}$$



Figure 5.7. Images of rose leaves.

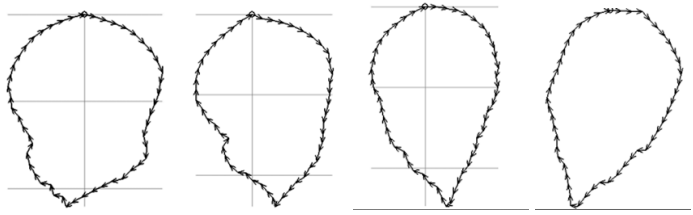


Figure 5.8. Realization of a random contour of the type "Rose" after parameters' correction.

The aggregate $\mathbf{X}_m$ for $m = -\infty, \infty$ defines the kind of closed continuous elementary contours (CEC). For $|\chi| = 1$, any two contours from the complete kind of continuous elementary contours are orthogonal,

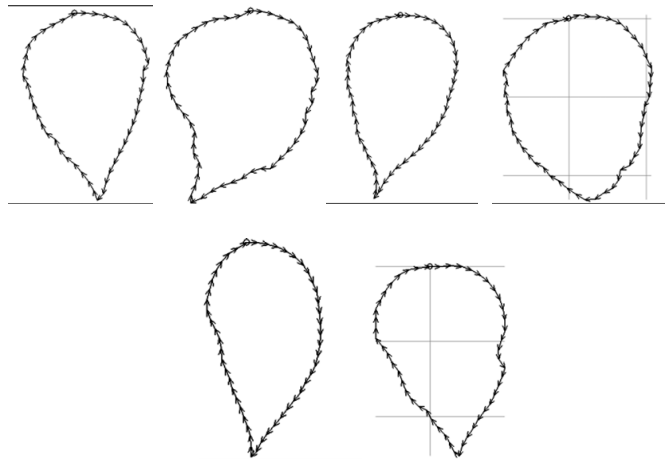


Figure 5.9. Ending.

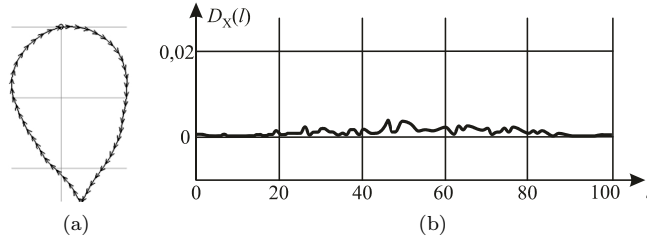


Figure 5.10. The contour of mathematical expectation (a) and the variance (b) of a random contour of the type "Rose".

that is, the scalar product of the contours is

$$(\mathbf{X}_m, \mathbf{X}_n) = \int_0^L \exp\left\{i\frac{2\pi}{L}(m-n)l\right\} dl = \begin{cases} L & \text{if } m = n, \\ 0 & \text{if } m \neq n. \end{cases}$$

For $|\chi| = 1/\sqrt{L}$, contours forming the basis are orthonormal:

$$(\mathbf{X}_m, \mathbf{X}_n) = \frac{1}{L}\int_0^L \exp\left\{i\frac{2\pi}{L}(m-n)l\right\} dl = \begin{cases} 1 & \text{if } m = n, \\ 0 & \text{if } m \neq n. \end{cases}$$

The elementary contours $\mathbf{X}_m$ и $\mathbf{X}_n$ preserve the orthogonality property when each of them rotates by random angles $\Delta\varphi_m$ and $\Delta\varphi_n$ and

extends $|\mu_m|$ and $|\mu_n|$ times respectively, i.e.

$$(\mu_m \mathbf{X}_m, \mu_n \mathbf{X}_n) = \begin{cases} |\mu_m|\,|\mu_n|\,\|\mathbf{X}\| & \text{if } m = n, \\ 0 & \text{if } m \neq n, \end{cases}$$

where $\mu_m = |\mu_m| \exp\{i\Delta\varphi_m\}, \mu_n = |\mu_n| \exp\{i\Delta\varphi_n\}$. Consequently, the orthogonality of the CEC is preserved by scaling and random rotating each of them.

Elementary continuous contours forming an orthonormal basis are periodic with period L. Set the value of the contour length from those reasons that the period of the first harmonic iteration in expression (5.4) for $m = 1$ is equal to 2π. Figure 5.11 shows elementary continuous contours for $m = 1$ and $m = 2$.

Obviously, the higher-order basis functions are circles with the number of turns equal to the harmonic number.

5.3.1.2 *Random continuous contours decomposition*

The spectrum $\mathbf{P} = \{\rho(m)\}_{-\infty,\infty}$ of a continuous contour $\mathbf{X} = \{\chi(l)\}_{0,L}$ is defined as

$$\begin{aligned} \rho(m) &= \int_0^L \chi(l) \exp\left\{-i\frac{2\pi}{L}ml\right\} dl \\ &= \int_0^L |\chi(l)| \exp\left\{-i\left(\frac{2\pi}{L}ml - \varphi(l)\right)\right\} dl, \\ m &= 0, \pm 1, \pm 2, \ldots. \end{aligned} \tag{5.5}$$

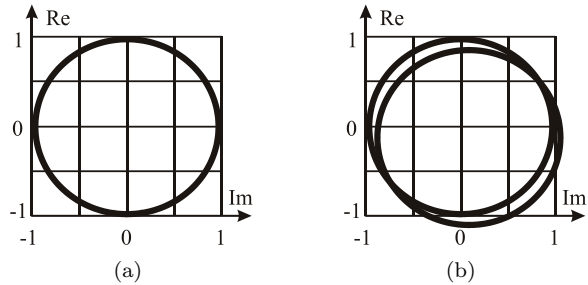


Figure 5.11. Elementary contours for $a - m = 1; b - m = 2$ (for clarity, the individual turns are spaced relative to each other).

Get the periodic contour of the Fourier series by calculating the decomposition coefficients $\mathbf{X} = \{\chi(l)\}_{0,L}$:

$$\chi(l) = \frac{1}{L} \sum_{m=-\infty}^{\infty} \rho(m) \exp\left\{ i \frac{2\pi}{L} ml \right\}. \tag{5.6}$$

If the decomposable signal is real, i.e. $\chi(l) = \mathrm{Re}\chi(l)$, then its spectrum $\mathbf{P} = \{\rho(m)\}_{-\infty,\infty}$, which is defined by expression

$$\rho(m) = \int_0^L \mathrm{Re}\chi(l) \exp\left\{ -i \frac{2\pi}{L} ml \right\} dl, \quad m = 0, \pm 1, \pm 2, \ldots$$

has the conjugate symmetry property (Figure 5.12(a))

$$\rho(m) = \rho^*(-m); |\rho(m)| = |\rho(-m)|.$$

Therefore, in symmetric points $m = 0$, the transformation coefficients' modules are the same

$$\begin{aligned} \rho(-m) &= \int_0^L \mathrm{Re}\chi(m) \exp\left\{ -i \frac{2\pi}{L} (-m) l \right\} dl \\ &= \int_0^L \mathrm{Re}\chi(l) \exp\left\{ i \frac{2\pi}{L} (-m) l \right\} dl. \end{aligned}$$

As $\mathrm{Re}\chi(l)$ is a real number, the change of sign of the exponent led to the fact that the complex number $\mathrm{Re}\chi(l) \exp\left\{ -i \frac{2\pi}{L} ml \right\}$

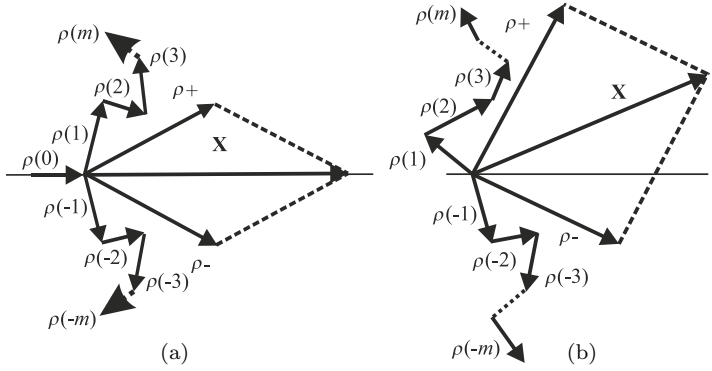


Figure 5.12. Graphical representation of the Fourier series in complex form for (a) real signal; (b) complex-valued signal.

was replaced by a complex conjugate $\mathrm{Re}\chi(l)\exp\left\{i\frac{2\pi}{L}ml\right\}$. Therefore, $\rho(m) = \rho^*(-m)$ and the real signal spectrum is characterized by symmetrically arranged relatively indication $m = 0$ complex-conjugate pairs and they are limited to half of the counts taken to the right of the point $m = 0$ when operating with the real signal amplitude spectrum (Figure 5.13(a)).

When the value $\chi(l)$ becomes a complex number, then to replace the term $\chi(l)\exp\left\{-i\frac{2\pi}{L}ml\right\}$ with the complex conjugate value, one must change the sign not only in the exponent but also in the argument $\chi(l)$. Thus complex-valued signals don't possess the conjugate symmetry property (Figure 5.12(b)) and it is necessary to consider all indications in their amplitude spectrum (for both positive and negative frequencies) (Figure 5.13(b)).

If the real $\mathrm{Re}\chi(l)$ and imaginary $\mathrm{Im}\chi(l)$ components of a continuous contour $\mathbf{X} = \{\chi(l)\}$ are considered as real functions, then their spectra $\rho_{\mathrm{Re}\chi(l)}(l)$ and $\rho_{\mathrm{Im}\chi(l)}(l)$ will have the symmetry property (complex conjugacy) relative to the ordinate axis. It is possible to get the spectrum of a continuous contour $\mathbf{X}= \{\chi(l)\}$ by summing the spectra of the function $\mathrm{Re}\chi(l)$ and $\mathrm{Im}\chi(l)$.

5.3.1.3 *Continuous contour spectrum properties*

Let us show that the contour $\mathbf{P}$ of the contour spectrum $\mathbf{X}$ is always open and the zero component of the spectrum of the closed contour is always equal to zero. Using expression (5.6) for the inverse Fourier transform for $l = 0$, we obtain

$$\chi(0) = \frac{1}{L}\sum_{m=-\infty}^{\infty}\rho(m).$$

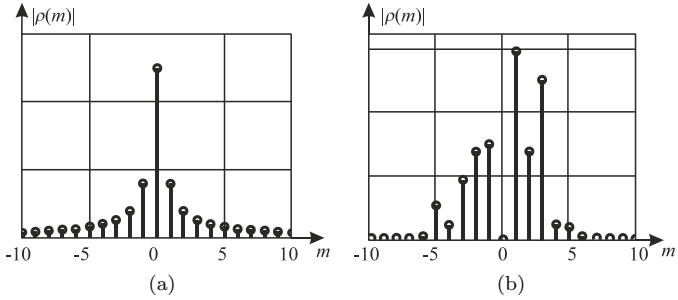


Figure 5.13. Amplitude–frequency spectra of (a) periodic sequence of sawtooth pulses; (b) complex-valued continuous contour.

As far as in general the instant value is $\chi(0) \neq 0$, then the sum of all elements of contour $\mathbf{P}$ is not equal to zero, i.e. it will be opened. Relying on (5.5) $m = 0$, we get

$$\rho(0) = \int_0^L \chi(l)dl.$$

The contour $\mathbf{X}$ is closed by condition, therefore $\rho(0) = 0$.

Let us pass to the complex conjugate contour by changing the signs in front of the imaginary part of each indication

$$\mathbf{X}^* = \{|\chi(l)\exp\{-i\varphi(l)\}|\}_{0,L}.$$

We define the form of the spectrum $\rho_{\mathbf{X}^*}(m)$, $m = 0, \pm1, \pm2, \dots$, of such a contour expressed in terms of the spectrum components $\rho(m)$ of the original contour $\mathbf{X}$:

$$\rho_{\mathbf{X}^*}(m) = \int_0^L \chi^*(l)\exp\left\{-i\frac{2\pi}{L}ml\right\}dl.$$

Then we will write the expression for the spectrum conjugate component of the conjugate contour

$$\begin{aligned}\rho^*_{\mathbf{X}^*}(m) &= \int_0^L \chi(l)\exp\left\{i\frac{2\pi}{L}ml\right\}dl \\ &= \int_0^L \chi(l)\exp\left\{-i\frac{2\pi}{L}(-m)l\right\}dl.\end{aligned}$$

This implies that $\rho^*_{\mathbf{X}^*}(m)$ is equal $\rho(-m)$. Then,

$$\rho_{\mathbf{X}^*}(m) = \rho^*_{\mathbf{X}^*}(-m), \quad m = 0, \pm1, \pm2, \dots.$$

Thus, the complex-conjugate continuous contour spectrum is being produced as a complex conjugate inverse Fourier transform of the original contour.

The starting point shift of the contour to a random number d leads to the multiplication of the spectrum by an exponential multiplier $\exp\left\{i\frac{2\pi}{L}md\right\}$, i.e.

$$\mathbf{P}^{(d)} = \left\{\rho(m)\ \exp\left\{i\frac{2\pi}{L}\ md\right\}\right\}_{-\infty,\infty}.$$

If the contour $\mathbf{Y}$ is a contour $\mathbf{X}$ subjected to a group of linear transformations: stretching, rotation, and the starting point shift,

the spectrum $\mathbf{P_Y}$ of the contour $\mathbf{Y}$ is related to the spectrum $\mathbf{P_X} = \{\rho_{\mathbf{X}}(m)\}_{-\infty,\infty}$ of the contour $\mathbf{X}$ as follows:

$$\mathbf{P_Y} = \left\{ |\mu| \exp\left\{ i \left[\frac{2\pi}{L} md + \Delta\varphi \right] \right\} \rho_{\mathbf{X}}(m) \right\}_{-\infty,\infty}.$$

5.3.2 *Continuous contours correlation analysis*

ACF and CCF continuous contours can be obtained based on the scalar product $(\mathbf{X},\mathbf{Y})$ of vectors $\mathbf{X} = \{\chi(l)\}_{0,L}$ and $\mathbf{Y} = \{\upsilon(l)\}_{0,L}$. By determination (5.3),

$$(\mathbf{X}, \mathbf{Y}) = \int_0^L \chi(l)\upsilon^*(l)dl.$$

To obtain CCF $\tau(r)$ contours $\mathbf{X}$ and $\mathbf{Y}$, we shall consider that these contours have the periodicity property (5.1). Then individual scalar product values can be obtained for each starting point position of the contour $\mathbf{X}$. The totality of the values of these scalar products form the CCF contours $\mathbf{X}$ and $\mathbf{Y}$:

$$\tau(r) = \int_0^L \upsilon(l)\chi^*(l+r)dl,$$

where r is any from 0 to L. When $\mathbf{X} = \mathbf{Y}$, the CCF passes to ACF:

$$\tau(r) = \int_0^L \chi(l)\chi^*(l+r)dl.$$

CCF allows us to determine the closeness degree of contours $\mathbf{X}$ and $\mathbf{Y}$ invariant to the angle of their mutual rotation, the starting point shift, and the scale.

ACF and CCF are complex functions and can be represented on a complex plane in the form of a contour $\mathbf{T} = \{\tau(r)\}_{0,L}$ (Figure 5.14).

The autocorrelation function (ACF) of the CEC $\mathbf{X}_m, m = 0, \pm1, \pm2, \ldots$, is the scalar product $\mathbf{X}_m$ and $\mathbf{X}_m^{(s)}$, s is any in the range from 0 to L, where $\mathbf{X}_m^{(s)}$ is the same CEC but with a starting point

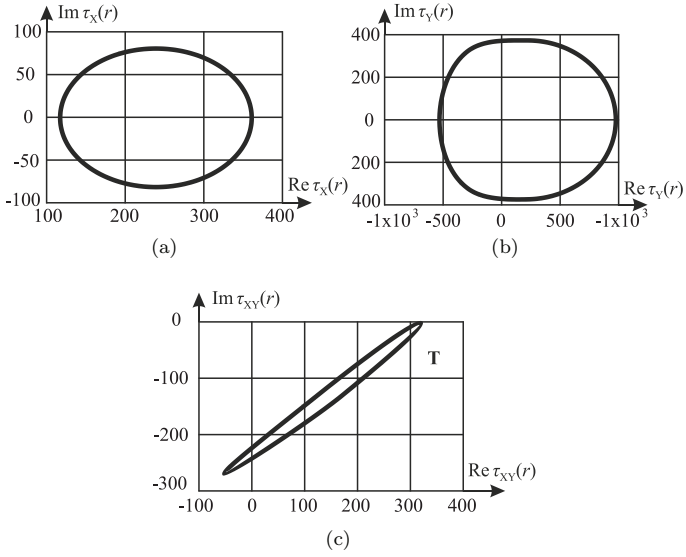


Figure 5.14. ACF contours **X** and **Y** (a), (b); their CCF (c).

shifted by s indications:

$$\tau_m(s) = (\mathbf{X}_m, \mathbf{X}_m^{(s)}) = \int_0^L \chi_m(l)\chi_m^*(l+s)dl$$
$$= \int_0^L \exp\left\{i\frac{2\pi}{L}ml\right\} \exp\left\{-i\frac{2\pi}{L}m(l+s)\right\} dl$$
$$= \int_0^L \exp\left\{-i\frac{2\pi}{L}ms\right\} dl = L\mathbf{X}_m^*.$$

Thus, the ACF $\mathbf{T}_m = \{\tau_m\}_{0,L}$ of the elementary contour $\mathbf{X}_m$ is a scale-modified and complex-conjugate contour $\mathbf{X}_m$.

The cross-correlation function (CCF) of a continuous elementary contour

$$\tau_{mn}(s) = (\mathbf{X}_m, \mathbf{X}_n^{(s)}) = \int_0^L \chi_m(l)\chi_n(l+s)dl$$
$$= \int_0^L \exp\left\{i\frac{2\pi}{L}ml\right\} \exp\left\{i\frac{2\pi}{L}n(l+s)\right\} dl$$
$$= \int_0^L \exp\left\{i\frac{2\pi}{L}ml\right\} \exp\left\{i\frac{2\pi}{L}nl\right\} \times \exp\left\{i\frac{2\pi}{L}ns\right\} dl$$

$$= \exp\left\{i\frac{2\pi}{L}ns\right\}\int_0^L \exp\left\{i\frac{2\pi}{L}l\,(m-n)\right\}dl.$$

In this expression, the integral differs from zero only when $m = n$. In this case, it is equal to L. Then,

$$\tau_{mn}(s) = \begin{cases} L\mathbf{X}_m^* & \text{if } m = n, \\ 0 & \text{if } m \neq n. \end{cases}$$

Thus, the CCF CEC everywhere equals zero, except for the case when $m = n$, i.e. when it passes into ACF.

Complex-valued signal nature, in comparison with the real signal, determines the different forms of ACF and CCF, since one of the factors under the integral sign is complex conjugate. Let us find the connection between the energy spectrum $\mathbf{W_X} = \{|\rho(m)^2|\}$ and the continuous contour ACF $\mathbf{T_X} = \{|r(l)^2|\}_{0,L}$. Now we replace the expression for the continuous contour ACF

$$\tau(s) = \int_0^L \chi(l)\,\chi^*(l+s)dl$$

instead of the contour indications of their Fourier transform

$$\chi(l) = \frac{1}{L}\sum_{m=-\infty}^{\infty}\rho(m)\exp\left\{i\frac{2\pi}{L}ml\right\},$$

$$\chi(l+s) = \frac{1}{L}\sum_{m=-\infty}^{\infty}\rho(m)\exp\left\{i\frac{2\pi}{L}m(l+s)\right\}.$$

Then,

$$\tau(s) = \int_0^L \left[\frac{1}{L}\sum_{h=-\infty}^{\infty}\rho(h)\exp\left\{i\frac{2\pi}{L}hl\right\}\right] \times \left[\frac{1}{L}\sum_{m=-\infty}^{\infty}\rho(m)\exp\left\{i\frac{2\pi}{L}m(l+s)\right\}\right]^* dl.$$

The complex conjugate product is equal to the product of complex conjugate factors. Considering this we transform the second square

bracket:

$$\tau(s) = \int_0^L \left[\frac{1}{L}\sum_{h=-\infty}^{\infty} \rho(h)\exp\left\{i\frac{2\pi}{L}hl\right\}\right] \times \left[\frac{1}{L}\sum_{m=-\infty}^{\infty} \rho^*(m)\exp\left\{-i\frac{2\pi}{L}ml\right\}\exp\left\{-i\frac{2\pi}{L}ms\right\}\right] dl.$$

Then, we regroup the addendums

$$\tau(s) = \frac{1}{L^2}\sum_{h=-\infty}^{\infty}\sum_{m=-\infty}^{\infty} \rho(h)\rho^*(m)\exp\left\{-i\frac{2\pi}{L}ms\right\} \times \int_0^L \exp\left\{i\frac{2\pi}{L}(h-m)l\right\} dl.$$

The integral in this expression is the sum of the CEC indicates about $h - m$. It does not equal to zero only when $h = m$. Therefore, the expression $\tau(s)$ will not be zero only if $h = m$, and in this case,

$$\tau(s) = \frac{1}{L^2}\sum_{m=-\infty}^{\infty} |\rho(m)|^2 \exp\left\{-i\frac{2\pi}{L}ms\right\}.$$

Thus, the continuous contour ACF is the result of the Fourier transform of its energy spectrum. A similar result is also true for a real signal, but its derivation is based on the conjugate symmetry properties of the spectrum which complex-valued signals do not possess.

5.3.3 *Random contours correlation and spectral analysis*

For two fixed values of the argument, for example, for $l = l_1$ and $l = l_2$, we obtain two sections: a system of two complex random variables $\mathrm{X}(l_1)$ and $\mathrm{X}(l_2)$ with a correlation moment $M\left[\overset{\circ}{\mathrm{X}}(l_1)\overline{\overset{\circ}{\mathrm{X}}(l_2)}\right]$. $\mathrm{X}(l_2)$ here is a contour that is complex conjugate to the contour $\overset{\circ}{\mathrm{X}}(l_2)$. The correlation function of a random complex contour $\mathrm{X}(l_1)$ is a non-random function $\mathrm{K}(l_1, l_2)$ of two independent arguments l_1 and l_2 whose value for each pair of fixed values of the arguments

is equal to the correlation moment of sections corresponding to the same fixed values of the arguments:

$$\mathrm{K_X}(l_1, l_2) = M\left[\overset{\circ}{\mathrm{X}}(l_1)\overline{\overset{\circ}{\mathrm{X}}(l_2)}\right].$$

For equal values of the arguments $l_1 = l_2 = l$, the random contour correlation function is equal to the variance of this contour:

$$\mathrm{K_X}(l, l) = D_{\mathrm{X}}(l). \tag{5.7}$$

If the real random functions Re X(l) and Im X(l) are correlated, then

$$\begin{aligned}\mathrm{K_X}(l_1, l_2) = {} & \mathrm{K_{ReX}}(l_1, l_2) + \mathrm{K_{ImX}}(l_1, l_2) \\ & + i[R_{\mathrm{ReX\ ImX}}(l_2, l_1) - R_{\mathrm{ReX\ ImX}}(l_1, l_2)],\end{aligned}$$

where $\mathrm{K_{ReX}}(l_1, l_2)$ and $\mathrm{K_{ImX}}(l_1, l_2)$ are the correlation functions of the real and imaginary parts of the random complex contour X(l) and $R_{\mathrm{ReX\ ImX}}(l_2, l_1)$ and $R_{\mathrm{ReX\ ImX}}(l_1, l_2)$ are the cross-correlation functions of the real and imaginary parts of the random complex contour X(l) under different method of following arguments.

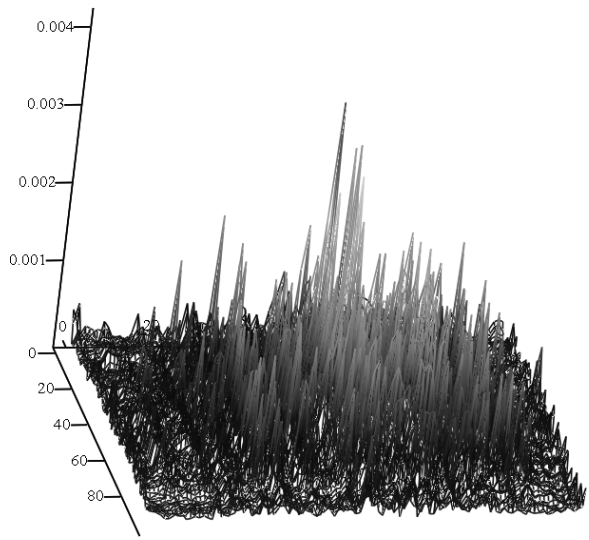


Figure 5.15. Random contour correlation function of the "Rose" type.

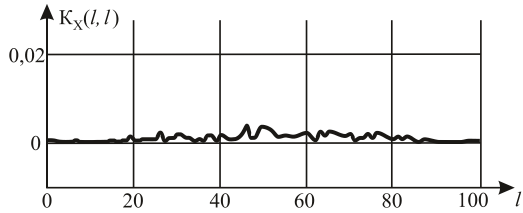


Figure 5.16. The random contour correlation function of the "Rose" type with equal arguments values.

If Re X(l) and Im X(l) are not correlated, then

$$\mathrm{K_X}(l_1, l_2) = \mathrm{K_{ReX}}(l_1, l_2) + \mathrm{K_{ImX}}(l_1, l_2).$$

The normalized correlation function $\rho_X(l_1, l_2)$ of a random contour X(l) is a non-random function of two independent variables l_1 and l_2, whose value for each fixed pair of values of the arguments is equal to the cross-section correlation coefficient corresponding to the same fixed values of the arguments:

$$\rho_{\mathrm{X}}(l_1, l_2) = \frac{\mathrm{K_X}(l_1, l_2)}{D_{\mathrm{X}}(l_1)\, D_{\mathrm{X}}(l_2)}.$$

The cross-correlation function of two random complex contours X(l) and Y(l) is the non-random function

$$R_{\mathrm{XY}}(l_1, l_2) = M\left[\overset{\circ}{\mathrm{X}}(l_1)\overline{\overset{\circ}{Y}(l_2)}\right].$$

The cross-correlation function of two random complex contours is expressed in terms of the cross-correlation functions of their real and imaginary parts as follows:

$$\begin{aligned} R_{\mathrm{XY}}(l_1, l_2) = {} & R_{\mathrm{ReX\ ReY}}(l_1, l_2) + R_{\mathrm{ImX\ ImY}}(l_1, l_2) \\ & + i[R_{\mathrm{ImX\ ReY}}(l_2, l_1) - R_{\mathrm{ReX\ ImY}}(l_1, l_2)]. \end{aligned}$$

Figure 5.15 shows the random contour correlation function K(l_1, l_2) of the "Rose" type.

In accordance with expression (5.7), the random contour correlation function $\mathrm{K_X}(l, l)$ of the "Rose" type is equal to the contour variances with equal values of the arguments $l_1 = l_2 = l$ (Figure 5.16).

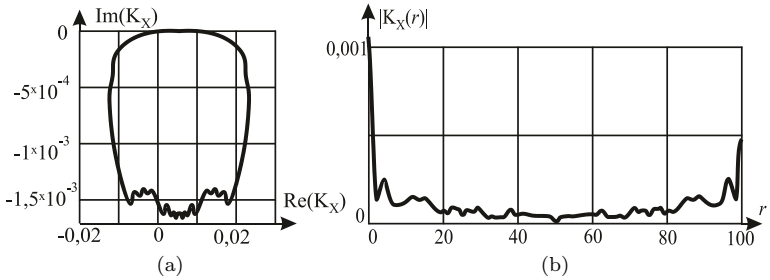


Figure 5.17. The contour (a) and the dependence of the correlation function modulus (b) for a random contour of the "Rose" type.

If the random contour $\mathrm{X}(l)$ is stationary, i.e. the contour expected value is constant for all argument values l and the correlation function depends only on the arguments difference $l_2 - l_1$, then the correlation function is a one-argument function $r = l_2 - l_1$:

$$\mathrm{K_X}(l_1, l_2) = \mathrm{K_X}(l_2 - l_1) = \mathrm{K_X}(r).$$

In this case, the variance of a stationary random contour is constant for all argument values l and equal to the value of its correlation function at the origin when $r = 0$:

$$D_{\mathrm{X}}(l) = \mathrm{K_X}(l, l) = \mathrm{K_X}(l - l) = \mathrm{K_X}(0).$$

Figure 5.17 shows the correlation function contour and the dependence plot of the correlation function modulus on the value r for a random contour of the "Rose" type. In this case, the quantity $\mathrm{K_X}(0) = 1,055 \cdot 10^{-3}$.

As shown in [5], the stationary random function can be represented in the harmonic oscillations form with random amplitudes and random phases:

$$X(l) = \sum_{n=1}^{\infty} [U_n \cos \omega_n l + V_n \sin \omega_n l] = \sum_{n=1}^{\infty} \sqrt{U_n^2 + V_n^2} \sin(\omega_n l + \varphi_n), \tag{5.8}$$

where $\varphi_n = \mathrm{arc}tg(U_n/V_n)$ and U and V are uncorrelated random values with zero expected value and the same variances.

The correlation function of sum (5.8) can be represented in the form [5]

$$\mathrm{K_X}(r) = \sum_{n=1}^{\infty} D_n \sin(\omega_n r + \varphi_n).$$

Considering that the random contour $\mathrm{X}(l)$ is periodic with period $L, \omega_n = \frac{2\pi n}{L}$. Then,

$$\mathrm{K_X}(r) = \sum_{n=1}^{\infty} D_n \sin\left(\frac{2\pi n}{L} r + \varphi_n\right). \tag{5.9}$$

The function $\mathrm{K_X}(r)$ will also be periodic with period L. In addition, the function $\mathrm{K_X}(r)$ is a complex-valued function.

The relation (5.9) can be regarded as the correlation function expansion in a Fourier series with coefficients:

$$D_n = \frac{1}{L} \int_0^L \mathrm{K_X}(r) \sin\left(\frac{2\pi n}{L} r + \varphi_n\right) dr. \tag{5.10}$$

The spectrum of a stationary random contour $\mathrm{X}(l)$ is the set of variances of all the harmonics components. In this case, the variance of one n-nd harmonic is equal to the variance of the quantity U_n (or the random variable variance V_n) considering that the random variables U_n and V_n are not correlated and the variances of the quantities with the same indices are equal. In this case,

$$D[\mathrm{X}(l)] = \sum_{n=1}^{\infty} D_n, \tag{5.11}$$

i.e. the stationary random contour variance can be represented as a sum of harmonics and is equal to the variance of its harmonics. Figure 5.18 shows the random contour spectrum of the "Rose" type.

The sum of the harmonics variances D_n of the random contour spectrum of the "Rose" type is equal to $1,055 \cdot 10^{-3}$, which corresponds to expression (5.11). In this case, the spectrum is only real. This is explained by the fact that the correlation function entering expression (5.10) of a random contour of the "Rose" type is conjugate

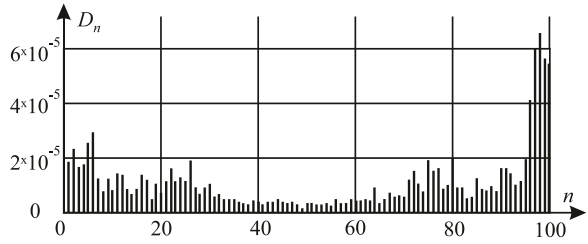


Figure 5.18. Random contour spectrum of the "Rose" type.

symmetric. The instantaneous values of the conjugate-symmetric contour are connected by a condition [1]:

$$\chi(l) = \chi^*(-l) = \chi^*(L - l).$$

The instantaneous value $\chi(0)$ of the conjugate-symmetric contour is only real. The conjugate-symmetric contours are symmetric relative to the vertical axis (see Figure 5.17(a)).

As shown in [1], the conjugate-symmetric contour spectrum is only real.

5.4 Discretization of a Continuous Contour

If $\mathbf{X} = \{\chi(l)\}_{0,2\pi}$ is an arbitrary contour whose spectrum $\mathbf{P} = \{\rho(m)\}_{-m_0,\, m_0}$ differs from zero only in the frequency line given by the numbers — $m_0 \le m \le m_0$, then it can be expanded into a generalized Fourier series in the Kotelnikov basis. Taking into account that the function $\mathbf{X}$ is complex-valued, we perform discretization separately for the real and imaginary parts. We get

$$\chi(l) = \sum_{n=-\infty}^{\infty} c_{1,\,n}\, Sc_n(l;\, m_0) + i\, c_{2,\,n}\, Sc_n(l;\, m_0),$$

where $c_{1,n} = (\mathrm{Re}\chi(l), Sc_n(l; m_0))$ and $c_{2,n} = (\mathrm{Im}\chi(l), Sc_n(l; m_0))$ are the coefficients of the series. Moreover, the quantities $c_{1,n}$ and $c_{2,n}$ are just the same as $\mathrm{Re}\gamma(n) = \mathrm{Re}\chi(l_n)$ and $\mathrm{Im}\gamma(n) = \mathrm{Im}\chi(l_n)$, i.e. the instantaneous values of the continuous contour $\chi(l)$ at the n-th point $l_n = n\pi/m_0$ [12].

Figure 5.19 shows an example of the process of an order $m = 1$ continuous elementary contour $\mathbf{X}_m = \{\chi_m(l)\}_{0,2\pi} =$

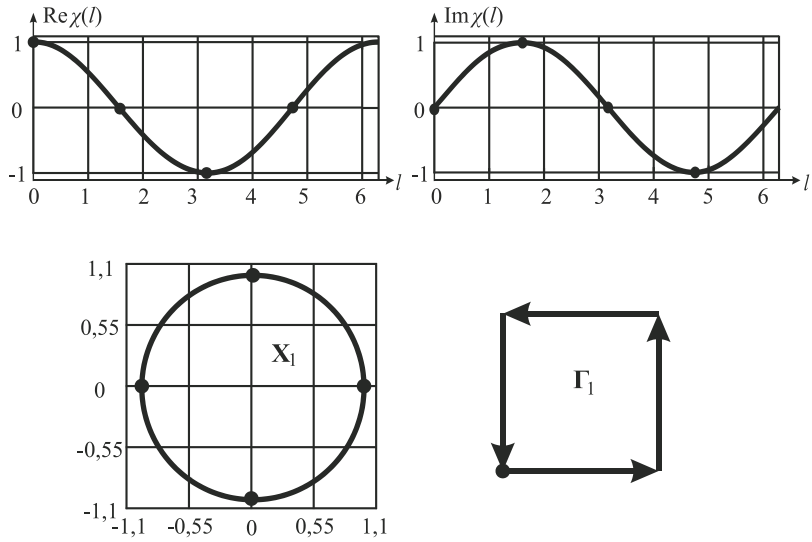


Figure 5.19. The process of discretization of a continuous elementary contour $\mathbf{X}_1$.

$\{|\chi|\exp\{iml\}\}_{0,2\pi}$ discretization. As a result of the discretization of a continuous elementary contour $\mathbf{X}_1$ of order $m = 1$, we obtain a discrete elementary contour $\mathbf{\Gamma}_1$ of order $m = 1$.

Figure 5.20 shows examples of continuous contours $\mathbf{X}$ and $\mathbf{Y}$ and their amplitude spectra.

Figures 5.21 and 5.22 show the contours obtained by sampling continuous contours $\mathbf{X}$ and $\mathbf{Y}$ at frequencies with numbers $m_0 = 5$ and 10.

5.5 Realization of Algorithms for Continuous Image Contours Processing

5.5.1 *Given shape image recognition*

The recognizing image problem of a given shape was solved in the following formulation. The alphabet of reference images contours from M classes is set. Represented in the contour description form the recognizable image is distorted by the noise effect and the linear distortions influence (changing the scale and angle of rotation). The recognition device must ensure the best solution acceptance for the

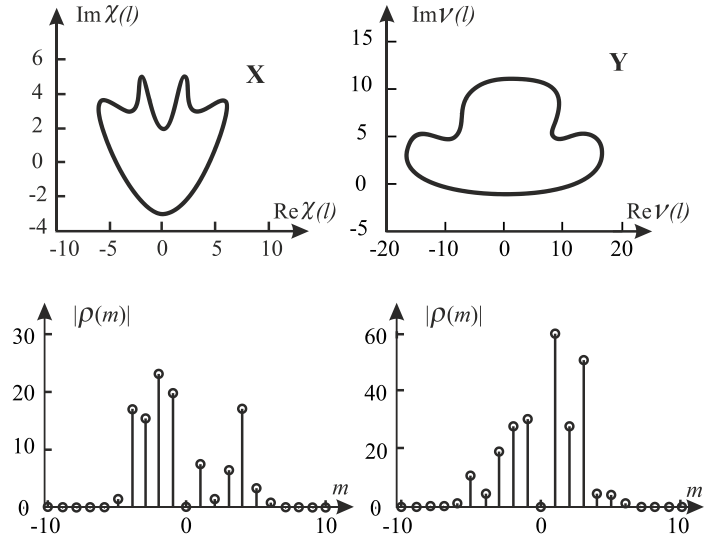


Figure 5.20. Examples of continuous contours and their amplitude spectra.

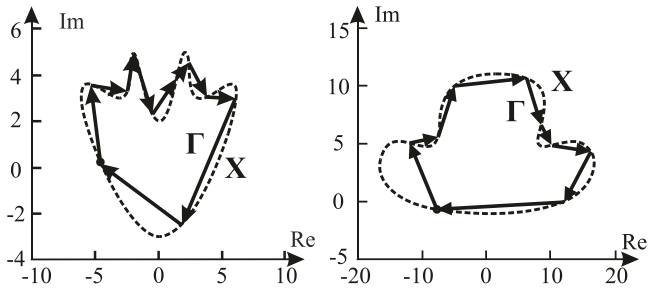


Figure 5.21. Contours obtained by discretizing a continuous contour at frequencies with numbers m_0 : (a) 5; (b) 10.

observed image class. Supposing that the solution is necessarily taken in favor of one of the M alphabet classes, i.e. there are no situations connected with the recognition rejection, then the acceptance solution can be considered as a two-alternative: the image is recognized correctly or the solution is wrong. In this case, the recognition device efficiency is characterized by the recognition probability P_{rp}.

Let us consider the operation of the recognizing algorithm for noisy continuous image contours by the example of two classes. Let $\mathbf{X}_{(1)}^{(c)} = \{\chi_{(1)}(l)\}$ and $\mathbf{X}_{(2)}^{(c)} = \{\chi_{(2)}(l)\}$ be continuous contours of classes

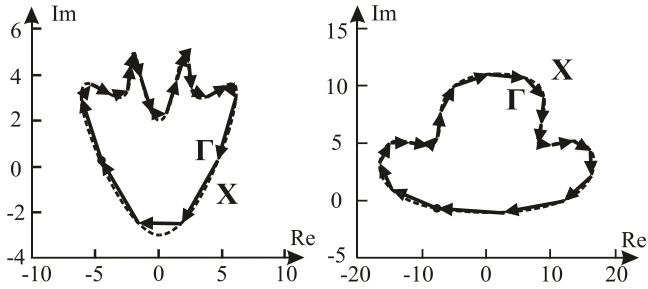


Figure 5.22. Ending.

A_1 and A_2. The noisy contour of the first class, which is equal to $\mathbf{Y} = \mathbf{X}_{(1)}^{(c)} + \mathbf{Z}$, or of the second class, equal to $\mathbf{Y} = \mathbf{X}_{(2)}^{(c)} + \mathbf{Z}$, where $\mathbf{Z}$ is the noisy contour, is supplied on the recognition device and

$$\mathbf{X}_{(1)}^{(c)} = |\mu_{(1)}| \exp\{i\Delta\varphi_{(1)}\}\{\chi_{(1)}(l + d_{(1)})\},$$

$$\mathbf{X}_{(2)}^{(c)} = |\mu_{(2)}| \exp\{i\Delta\varphi_{(2)}\}\{\chi_{(2)}(l + d_{(2)})\}.$$

Parameters $|\mu|, \Delta\varphi$, and d are expected to be unknown.

The recognition task is in the well-founded assignment of the noisy contour $\mathbf{Y}$ to one of the classes A_1 and A_2. To make a solution, it is necessary to form an unconditional likelihood ratio and compare it with the threshold value. The likelihood functions are

$$W(\mathbf{Y}, |\mu_{(1)}|, \Delta\varphi_{(1)}, d_{(1)}) = c \cdot \exp\left\{-\frac{1}{2\sigma^2}\left[\|\mathbf{Y}\|^2 + \left\|\mathbf{X}_{(1)}^{(c)}\right\|^2 - 2\mathrm{Re}\left(\mathbf{Y}, \mathbf{X}_{(1)}^{(c)}\right)\right]\right\},$$

$$W(\mathbf{Y}, |\mu_{(2)}|, \Delta\varphi_{(2)}, d_{(2)}) = c \cdot \exp\left\{-\frac{1}{2\sigma^2}\left[\|\mathbf{Y}\|^2 + \left\|\mathbf{X}_{(2)}^{(c)}\right\|^2 - 2\mathrm{Re}\left(\mathbf{Y}, \mathbf{X}_{(2)}^{(c)}\right)\right]\right\}.$$

Then the unconditional likelihood ratio logarithm will be written as

$$\ln[L(\mathbf{Y}, |\mu_{(1)}|, |\mu_{(2)}|, \Delta\varphi_{(1)}, \Delta\varphi_{(2)}, d_{(1)}, d_{(2)})] = -\frac{1}{2\sigma^2}\left\{\left\|\mathbf{X}_{(1)}^{(c)}\right\|^2 - \left\|\mathbf{X}_{(2)}^{(c)}\right\|^2 - 2\left[\mathrm{Re}\left(\mathbf{Y}, \mathbf{X}_{(1)}^{(c)}\right) - \mathrm{Re}\left(\mathbf{Y}, \mathbf{X}_{(2)}^{(c)}\right)\right]\right\}.$$

The recognition algorithm provides a multichannel procedure for the class number if a noisy contour from a certain alphabet is allocated to

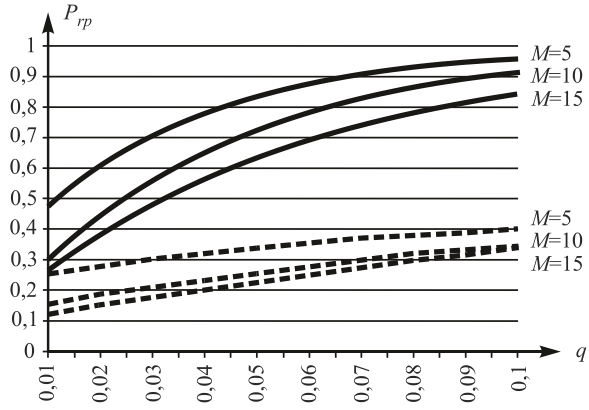


Figure 5.23. The probability of correct image recognition of the given shape objects in the circle form for volumes of the alphabet $M = 5, 10$, and 15.

the recognition device input. Each channel contains a filter matched to the reference contour of the corresponding class and the extreme device producing the instantaneous value with the maximum modulus. Further, the channel in which the instantaneous output value module at the output exceeds the output modules of the remaining channels is noted. If the maximum modulus value is greater than the threshold value, then the recognizable contour belongs to a class whose number is equal to the channel number.

Figure 5.23 shows the dependency plots of the probability of correct image recognition P_{rp} of the given shape objects in the circle form based on an analysis of their contours represented as a continuous line (solid) and discrete (dotted) from the signal-to-noise ratio for volumes of the alphabet $M = 5$, 10, and 15.

The value q is equal to the ratio of the elementary vector average norm $\|\mathbf{X}_m\|^2/(2\pi)$ of the continuous elementary contour $\mathbf{X}_m$ to the noise contour dispersion $\mathbf{Z}$: $q = \|\mathbf{X}_m\|^2/(2\pi\sigma^2)$.

The scalar product of two CEC orthogonal signals $\mathbf{X}_m$ and $\mathbf{X}_n$ is zero. Therefore, the distance between normalized CEC is constant. The decrease in the probability P_{rp} in the noisy continuous elementary contours recognition associated with the class number increase M is explained by the increase in the parallel channels number for each of which a wrong recognition decision can be made.

The question of the influence of image continuous contour discretization on the recognition characteristic is investigated. Figure 5.24 shows the dependency plots of the probability of correct

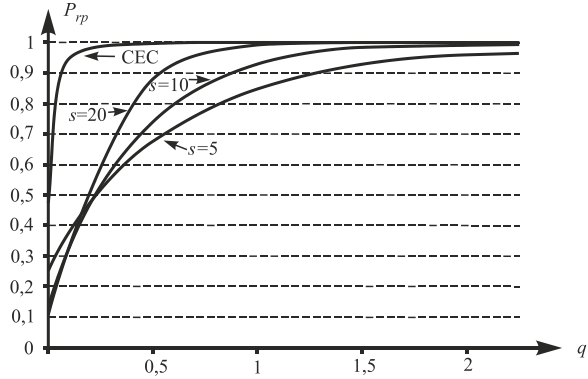


Figure 5.24. The probability of correct video image recognition of given shape objects in the circle form with a dimension $s = 5$, 10, and 20.

image recognition P_{rp} of given shape objects in the circle form from the signal-to-noise ratio q. Also, the contours were given in a discrete form with dimensions of $s = 5$, 10, and 20, and in the continuous line form.

It is obtained that, for example, with a signal-to-noise ratio $q = 0.5$ the probability of correct recognition P_{rp} with a contour dimension $s = 5$ is 0.67, at $s = 10, 0.73$, at $s = 20, 0.88$, and when the contour is set in the continuous line form, it is close to 1. It can be concluded that using continuous contours makes it possible to increase the probability of object of interest recognition, thereby increasing the noise immunity of the system.

Based on the research results, recommendations were developed on the discretization of the image contours of given shape objects in the circle form. Table 5.1 presents the minimum acceptable number of contour elements to provide a given recognition quality at a fixed signal-to-noise ratio q.

5.5.2 *The random contour detection*

The recognizing images with a random form task is solved in two stages. At the first stage, the contour image model is formed, and in the second stage, recognition is directly performed.

Let us consider the problem of contour images with a random form recognizing using the example of random contours

Table 5.1. The recommendations of the discretization parameter of image contours of given shape objects in the circle form.

			$q = 0.25$			
P_{np}	0.5	0.6	0.7	0.8	0.9	1
s	5	20	more 20	more 20	more 20	more 20
			$q = 0.5$			
P_{np}	0.5	0.6	0.7	0.8	0.9	1
s	5	5	10	20	more 20	more 20
			$q = 1$			
P_{np}	0.5	0.6	0.7	0.8	0.9	1
s	5	5	5	5	10	20

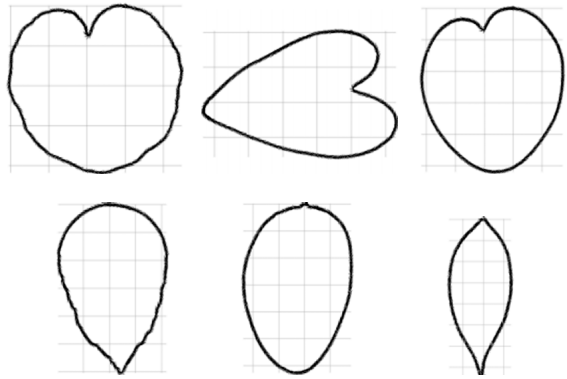


Figure 5.25. The expected value contours of the researched random contours classes.

$\mathbf{X}_1(l), \mathbf{X}_2(l), \ldots, \mathbf{X}_6(l)$ which characterize the classes "Rose", "Geranium", "Leaf", "Ficus1", "Ficus2", and "Cyclomen".

The expected value contours of these classes are shown in Figure 5.25.

The recognition device (Figure 5.25) provides an implementation $\chi(l)$ belonging to one of the classes. The recognition task is in the contour justified assignment $\chi(l)$ to one of the classes $A_1, A_2, \ldots, A_6$.

The device shown in Figure 5.26 is multichannel. Each channel contains a filter matched to the expected value contour of each class and a device that generates an instantaneous value with a maximum

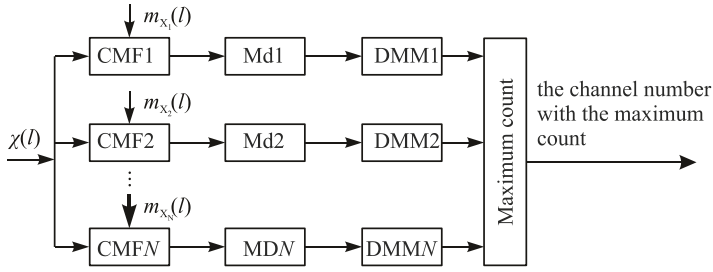


Figure 5.26. The recognition device structure: CMF — contour matched filter (MD — module driver, DDM — the device for determining the maximum).

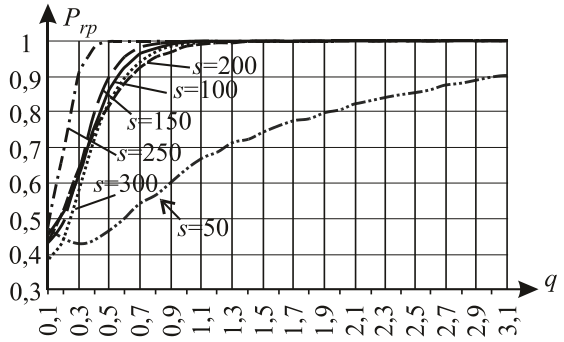


Figure 5.27. The probability of correct recognition P_{rp} of a noisy "Leaf" type contour from the signal-to-noise ratio for different contour dimensions.

modulus. After that, a channel, in which the output signal module exceeds the output modules of the remaining channels, is allocated. The recognizable contour belongs to the class whose number is equal to the channel number with the maximum count.

Figure 5.27 shows the dependency plots of the probability of correct recognition P_{rp} of a noisy "Leaf" type contour from the signal-to-noise ratio with the contour dimensions s of 50, 100, 150, 200, 250, and 300.

The increase in probability P_{rp} when noisy contours are recognized as their dimensionality s increases is explained by a more accurate image of contours representation.

Figure 5.28 shows the dependency plots of the probability of correct recognition P_{rp} of a noisy "Ficus 1" type contour from the signal-to-noise ratio for an alphabet volume M equal to 2, 3, 4, 5, and 6.

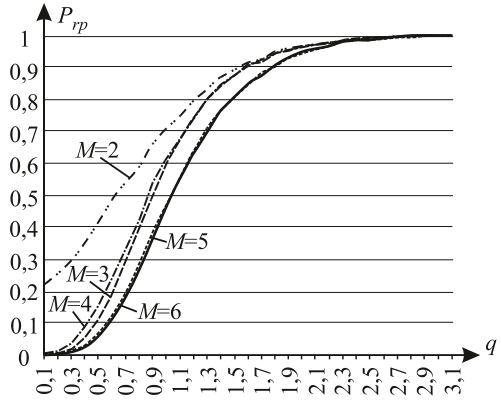


Figure 5.28. The probability of correct recognition P_{rp} of a noisy "Ficus 1" type contour from the signal-to-noise ratio for different alphabet volumes.

5.6 Conclusion

The advantage of contour analysis in the problems of image processing and visual image recognition lies in the possibility of solving these problems from the unified theoretical positions of the signal theory. The contours of the images completely characterize their shape and allow us to create simple analytical descriptions, invariant to the shift, rotation, and scale. Consideration of the image contours as complex-valued signals and their representation in a linear complex-valued space makes it possible to obtain a measure of two contours likeliness in the form of their scalar product, which is invariant to shift, rotation, and scale transforms. In this case, it is possible to estimate the rotation angle and the scale factor from the value of the scalar product.

An approach to the description of continuous complex-valued signals represented on the complex plane in the form of closed contours is considered. A linear space of vector contours is defined and the main analytic relations are obtained. The problems of spectral and correlation analysis of continuous contours are considered and features of their spectra are described. The structure of the continuous image contour processing device and the results of its modeling are presented.

Acknowledgments

This study was supported by the Russian Science Foundation grant № 23-19-00568 "Methods and intelligent system for supporting dynamic stability of operators of ergatic systems".

References

[1] Furman, Y.A. *et al.* (2002) *Introduction to Contour Analysis and Its Application to the Processing of Images and Signals.* In Furman, Y.A. (ed.) (Fizmatlit, Moscow).
[2] Furman, Y.A. *et al.* (2004) *Complex-Valued and Hypercomplex Systems in Multidimensional Signal Processing Problems.* In Furman, Y.A. (ed.) (Fizmatlit, Moscow).
[3] Furman, Y.A. *et al.* (2007) *Visualization of Images in Three-Dimensional Scenes* (MarGTU, Yoshkar-Ola).
[4] Gashnikov, M.V. *et al.* (2003) *Methods of Computer Image Processing*, 2nd Ed. In Soifer, V.A. (ed.) (Fizmatlit, Moscow).
[5] Gmurman, V.E. (2003) *Theory of Probability and Mathematical Statistics: A Textbook for Universities* (Higher School, Moscow).
[6] Gonzalez, R. and Woods, R. (2012) *Digital Image Processing* (Technosphere, Moscow).
[7] Khafizov, R.G. (2006) Analysis of continuous complex-valued signals defining the contours of images of flat objects. *Bulletin of KSTU Named after A.N. Tupolev*, 4, 24–27.
[8] Khafizov, R.G. and Okhotnikov, S.A. (2010) Linear filtration of continuous contours of images given in a complex-valued form. *Computer Optics*, 34(3), 408–416.
[9] Khafizov, R.G. and Okhotnikov, S.A. (2012) Recognition of continuous complex-valued image contours. *Izvestiya Vuzov. Instrument Making*, 5, 3–9.
[10] Khafizov, R.G. *et al.* (2013) Recognition of images with a random form. *Bulletin of the Volga State Technological University*, 2(18), 52–60.
[11] Khafizov, R.G. (2014) Model of a contour of the image with a random form. *Computer Optics*, 38(1), 127–130.
[12] Okhotnikov, S.A. and Khafizov, R.G. (2012) Discretization of continuous contours of images given in a complex-valued form. *Computer Optics*, 36(2), 274–278.

© 2025 World Scientific Publishing Company
https://doi.org/10.1142/9789811267215_0006

Chapter 6

Modeling and Processing of Image Sequences

Konstantin K. Vasiliev[*,‡], **Viktor R. Krasheninnikov**[*,§], **and Vitaliy E. Dementev**[†,¶]

[*] *Ulyanovsk State Technical University, Ulyanovsk, the Russian Federation*
[†] *National Research Mordovia State University, Saransk, the Russian Federation*
[‡] *vkk@ulstu.ru*
[§] *kvrulstu@mail.ru*
[¶] *vitawed@mail.ru*

Mathematical modeling and development of statistical algorithms for multidimensional image processing are examined. Optimal and quasi-optimal solutions of the main image processing tasks (e.g. filtration, detection, recognition, and alignment) are also described. The authors suggest adaptive processing algorithms with pseudo-gradient procedures.

Keywords: Multidimensional image model, random field, image processing, prediction, filtration, anomaly detection, recognition, alignment, potential efficiency, fixed point, adaptive algorithm, pseudo-gradient.

6.1 Introduction

Problems of automatic image analysis and their sequence analysis are becoming more and more important [1–53]. This is due to the

rapid development of aerospace Earth monitoring systems, radio and sonar systems with spatial arrays, medical devices for early disease diagnosis, computer vision systems, etc.

For some applied problems image can be represented as a data array set on a multidimensional index grid changing in discrete time. This allows us to describe certain 2D or multidimensional images and their temporal sequences with scalar or vector values, e.g. sequence of television images, multispectral images, and spatial field of wind speeds.

Mathematical models are necessary to formalize image processing tasks. Random nature of image values determines the use of probability theory and mathematical statistics methods, i.e. image representation in the form of random fields (RF). This implies analysis and synthesis problems of RF models.

Composition of information and noise RF can be represented as a model of image observations. This model can also include an additional parameter vector, which allows us to consider the peculiarities of image registration, e.g. possible mutual spatial displacement of adjacent frames and various abnormalities (object and signal).

This chapter presents the results obtained by the authors and their colleagues in modeling and statistical methods of multidimensional image and their sequence processing.

Section 6.2 covers the problems connected with random field description and simulation. Random fields are set on multidimensional spatial grids with rectangular or more complex cells. A number of models including multidimensional ones [3–16,37,38,47–49] can be used to describe the images. This chapter presents autoregressive, tensor, and wave models [18–20, 40–43]. It is difficult to obtain RF with spatial isotropic correlative bonds. It can be achieved by using models with multiple roots of the characteristic equation [42,43]. For linear transformations of image frames, it is desirable to use tensor operations, which provide the basis for tensor models [43]. Analysis and synthesis problems can be solved rather simply for wave models, in which RF is the result of disturbance interaction. These disturbances can occur in random places at random times [18, 22, 25]. Autoregressive and wave models allow us to describe images on different surfaces [18]. For example, unfolding of a cylindrical image can be used to represent speech signals [19], and the image on the sphere can be used to represent and simulate the lunar relief.

There are many papers on image filtering. In Section 6.3, the obtained models of RF are used to obtain tensor algorithm [43] and asymptotic characteristics of random field filtration efficiency on arbitrary dimension grids [44].

Problems of image and signal detection and recognition are discussed in a variety of works on random processes, 2D and stereo images, in particular [4, 5, 31, 34, 47, 51, 53]. Section 6.4 is devoted to the synthesis and analysis of optimal decision rules of point and lengthy anomaly detection on images and their sequences [22, 23]. It is shown that there are four equivalent forms for optimal procedures.

An important task of image sequence processing is their alignment, when due to various reasons there are spatial deformations of adjacent frames. There exist a number of approaches on how to solve this problem for 2D images, such as search for characteristic points, light flow analysis, morphological analysis, and correlation-extreme methods. In Section 6.5, the problems of alignment and estimation of parameters of image inter-frame geometrical transformations are considered for multidimensional image sequences. Various approaches to solving this problem are described, including tensor shift filtration and the method of a fixed point [21, 23, 43].

Section 6.6 outlines the adaptive procedure options, which are included in the algorithms for solving such problems as signal detection, filtering, and image alignment. Special attention is paid to the pseudo-gradient adaptation [43], on the basis of which highly efficient algorithms with comparatively small computational costs are synthesized. These qualities allow their usage in real-time systems, working with large images and their sequences.

6.2 Mathematical Models of Images

6.2.1 *A review of existing approaches to the description and processing of multidimensional images*

A large number of works related to the processing of various kinds of images and their sequences have been published in the world of image processing literature over the past years. At the same time,

the problems of modeling multidimensional arrays of correlated data and their optimal and suboptimal processing based on the proposed models have been considered in a very limited number of works. In the main, world and Russian specialists in image processing tend to use neural networks and large data [6, 17, 28, 36], which is certainly a promising direction of research development, but it does not allow to solve problems of understanding multidimensional images and forming optimal procedures for their processing. Among the methods for describing multidimensional images, the most promising are models of images defined on sequences of multidimensional grids [9, 33]. This representation of the images makes it simple enough to perform operations with the existing image models both element-by-element and as a whole object. However, the construction of such models by specifying probability distributions in general is quite difficult, therefore, various simplifications are used. The most important of these are the Markov property and homogeneity. The most suitable models for representing images with such properties are autoregressive models [2, 9]. However, such models, transformed to represent random fields on multidimensional grids, do not adequately describe many real images with heterogeneity.

An analysis of non-autoregressive models shows that most of them are associated with significant difficulties in identifying characteristics, although they allow the formation of heterogeneous images. Among them, we should distinguish a wave model [18] and Gibbs random fields [26, 38]. The formation of such fields often requires a sufficiently large number of iterations to provide the specified properties, but a more significant disadvantage is the complexity of the analytical description and, as a consequence, the problems of the development and technical implementation of algorithms for processing large image sequences.

The search for new probabilistic models of sequences of multizone images can be continued in the field of application of mixed models. For example, textures have been widely used in image processing [1, 33]. Their use is largely due to the wide range of images that can be picked up close enough to real images. In addition, aerospace observations have quite complex properties, which makes it difficult to analyze data only by spectral characteristics. Nevertheless, with the use of textures, it seems unlikely that a holistic representation of the actual image is possible. In this case, the model description is divided into several sections. The formation of random fields by

nonlinear transformations allows one to obtain non-Gaussian distributions. Mixed Gaussian models [35] and models based on the Gaussian bond and wavelet decomposition [50] give very successful results when working with known databases but do not have a sufficiently developed mathematical apparatus for their investigation.

Thus, we can assume that a complete solution to the problem of describing images to date has not been received. In this regard, the development and study of statistical models of images remains an urgent task.

6.2.2 *Random fields*

Nowadays, information systems including spatial sensor devices and digital computing are widely used. Therefore, we primarily consider images with discrete spatial and temporal variables. Without loss of generality, we may assume that the images are sets of values arranged on multidimensional rectangular grids with a unit step. Figures 6.1(a) and 6.1(b) show two-dimensional and three-dimensional grids. In general, the image is set in n-dimensional grid nodes $\Omega = \{\bar{j} = (j_1, \ldots, j_n) : j_k = \overline{1, M_k}, k = \overline{1, n}\}$.

According to the physical nature image values may be scalar (e.g. brightness of a monochromatic image), vector (velocity field, color images, and displacement field), and more complex (e.g. matrix). If we denote the image value in the node (pixel) $\bar{j}$ as $x_{\bar{j}}$ then the image is a set of these values on the grid: $X = \{x_{\bar{j}} : \bar{j} \in \Omega\}$.

If the data is a temporal sequence of images, then we can assume that the sequence is a single image by increasing the grid dimension by one unit. For example, the sequence of planar images (Figure 6.1(a)) can be regarded as a single three-dimensional image (Figure 6.1(b)).

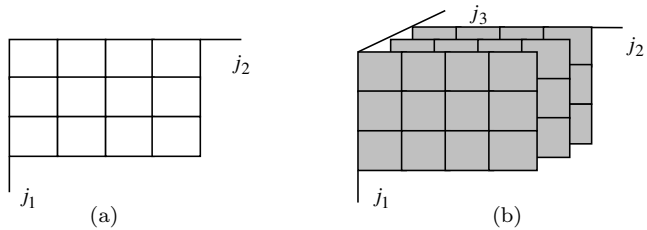


Figure 6.1. Grids with (a) two-dimensional image and (b) three-dimensional image.

If it is necessary to specify a time variable, let's set it down at the top: $X = \{x^i_{\bar{j}} : \bar{j} \in \Omega, i \in I\}$. This image is set on the direct product $\Omega \times I$ of grids Ω and I, where I is a set of time index values. The cross-section $x^i = \{x^i_{\bar{j}} : \bar{j} \in \Omega\}$, i.e. a set of image samples at a fixed value of time index i, is called the ith frame. Each frame is set on a grid Ω. For example, Figure 6.1(b) shows three two-dimensional frames.

Thus, an image can be determined as a function defined on a multidimensional grid. The values of the image elements cannot be accurately predicted in advance (otherwise monitoring system would not be necessary). Therefore, it is intrinsic to regard these values as random variables (RV) using the methods of probability theory and mathematical statistics.

6.2.3 *Tensor models of random fields*

Let's consider RF $X = \{x^i_{\bar{j}} : i \in I, \bar{j} \in \Omega\}$, defined on an $(n+1)$-dimensional grid $\Omega \times I$, where $\Omega = \{j = (j_1, j_2, j_3, \ldots, j_n)\}$ is an n-dimensional $M_1 \times M_2 \times \cdots \times M_n$-grid; $I = \{i : i = 1, 2, 3 \ldots\}$. Index i may be interpreted as a time index, so let's call cross-section $x^i = \{x^i_{\bar{j}} : \bar{j} \in \Omega\}$ of a field X as an ith frame.

Let the sequence of frames be described by a stochastic difference equation

$$x^i = \varphi^i(x^{i-1}) + \vartheta^i(x^{i-1})\xi^i, \quad i = 1, 2, \ldots, \tag{6.1}$$

where $\{\xi^i_{\bar{j}} : i \in I, \bar{j} \in \Omega\}$ is an updating standard Gaussian field, $\xi^i = \{\xi^i_{\bar{j}} : \bar{j} \in \Omega\}$ is the ith field frame, $\varphi^i(x^{i-1}) = \{\varphi^i_{\bar{j}}(x^{i-1}) : \bar{j} \in \Omega\}$ is $M_1 \times M_2 \times \cdots \times M_n$-matrix function, and $\vartheta^i(x^{i-1}) = \{\vartheta^i_{\bar{j}\bar{t}}(x^{i-1}) : \bar{j}, \bar{t} \in \Omega\}$ are tensors of rank $2n$ with group indices forming a perturbation component of the ith frame from ξ^i using the rule of tensor multiplication $\vartheta^i\xi^i = \{\vartheta^i_{\bar{j}\bar{t}}\}\{\xi^i_{\bar{t}}\} = \{\sum \vartheta^i_{\bar{j}\bar{t}}\xi^i_{\bar{t}}\}$. Transposing this frame supposes permutation of its group indices: $\vartheta^{iT}_{\bar{j}\bar{t}} = \vartheta^i_{\bar{t}\bar{j}}$. Note that the superscript i indicates the frame number, i.e. time in our case, so i is considered to be an umbral index and the summation with respect to it is not extended.

Model (6.1) allows us to describe a very broad class of Markov random frame sequences. In particular, a linear model

$$x^i = P^i x^{i-1} + \vartheta^i\xi^i, \tag{6.2}$$

where tensors P^i and ϑ^i do not depend on x^{i-1}, describes the Gaussian sequence. For this field of covariance functions (CF), there exists a multidimensional covariance matrix defined directly:

$$V_x(i,j) = M[(x^i - m^i) \times (x^j - m^j)], \quad V_x(i) = V_x(i,i),$$

where $m^i = M[x^i] = \{M[x^i_{\bar{j}}] : \bar{j} \in \Omega\}$ and symbol $\langle\times\rangle$ indicates external matrix multiplication. Thus, $V_x(i)$ and $V^i_\xi = M[\xi^i \times \xi^i]$ are symmetrical $M_1 \times M_2 \times \cdots \times M_n \times M_1 \times M_2 \times \cdots \times M_n$-matrices. For a complete definition of a random field with a state (6.2), it is necessary to set the law of first frame x^0 distribution. It is often a Gaussian distribution with mean m^0 and covariance matrix V^0_x.

Model (6.2) with constant tensors $P^i = P$ and $\vartheta^i = \vartheta$ is of particular interest:

$$V_x(i, i+k) = P^{(k)} V_x(i,i) = P^{(k)} V_x(i), \tag{6.3}$$

where (k) determines raising to the kth power. If the roots of the characteristic equation $\det(\lambda E - P) = 0$ are less than 1 in absolute value, then $P^{(k)} \to 0$ when $k \to \infty$, and from (6.3), we obtain that $V_x(i, i+k) \to 0$ when $k \to \infty$.

Using the z-transformation device let's write (6.2) in the following form

$$x^i = zPx^i + \vartheta\xi^i$$

or

$$(E - zP)x^i = \vartheta\xi^i.$$

Multiplying the congruence on the left by $(E - zP)^{-1}$, we obtain the expression x^i through a perturbing field:

$$x^i = (E - zP)^{-1}\vartheta\xi^i.$$

Hence,

$$x^i(x^i)^T = (E - zP)^{-1}\vartheta\xi^i(\xi^i)^T\vartheta^T(E - z^{-1}P^T)^{-1},$$

that after averaging gives the expression of the tensor spectrum for a stationary field

$$V_x(z) = \sum_{k=-\infty}^{\infty} V_x(0,k)z^k = (E - zP)^{-1}\vartheta V_\xi \vartheta^T (E - z^{-1}P^T)^{-1},$$

which is a Laurent series in powers of z with tensor coefficients. These coefficients can be found from

$$V_x(0,k) = \frac{1}{2\pi i}\int_{C_1}(E - zP)^{-1}\vartheta V_\xi \vartheta^T (E - z^{-1}P^T)^{-1} z^{k-1} dz, \quad (6.4)$$

where $C = \{z : |z| = 1\}$ is a unit circumference of a complex plane.

It is sufficient to find $V_x = V_x(0,0)$ from (6.4), and other values are obtained from (6.3), which in a stationary case looks like $V_x(i, i+k) = P^{(k)}V_x$. To find V_x it is possible to use limit (6.3) when $i \to \infty$ instead of integral in (6.4):

$$V_x = PVP^T + \vartheta V_\xi \vartheta^T.$$

This equation is a non-singular system of linear equations relative to tensor components V_x.

Now let's consider a solution to the model synthesis problem (6.2), i.e. the problem of finding tensors P^i and ϑ^i when intra-frame $V_x(i,i)$ and inter-frame $V_x(i-1,i)$ covariance tensors are given. From (6.3), we obtain

$$V_x(i-1,i) = P^i V_x(i-1, i-1),$$

which is a system of linear equations for tensor elements P^i. It is obvious that

$$P^i = V_x(i-1,i)V_x^{-1}(i-1,i-1).$$

Let's choose a perturbing field with covariance $V_\xi^i = E$, then we obtain the equation

$$\vartheta^i \vartheta^{iT} = V_x(i,i) - P^i V_x(i-1,i-1)P^{iT}.$$

For example, for tensor components ϑ^i, this equation can be solved on the basis of Gram–Schmidt orthogonalization.

Nonlinear stochastic difference equation may be considered as the generalization of the considered tensor model (6.2). This equation allows us to describe a very large class of Markov non-Gaussian RF on n-dimensional grids J_t. Here we have $\{\xi_{\bar{l}}^t, \bar{l} \in J_t, t \in T\}$, a field of independent, generally speaking, non-Gaussian random variables with known probability density function (PDF) $W(\xi_{\bar{l}}^t)$; $\varphi_{\bar{j}}^t(x_{\bar{l}}^{t-1})$ and $\vartheta_{\bar{j}\bar{l}}^t(x_{\bar{s}}^{t-1})$ are tensors of ranks n and $2n$ correspondingly, which in a general case nonlinearly depend on the values $\{\xi_{\bar{j}}\}$ of the (t–1)th frame in a multidimensional RF $\{x_{\bar{j}}^{t-1}, t \in T, \bar{j} \in J\}$.

6.2.4 *Autoregressive models of random fields*

In their structure, random fields are much more complex than stochastic processes. First, implementations of RF are functions of several variables, the theory of which is more complicated than of one variable. Second, the concept of Markov behavior also becomes much more complicated. A random process can develop in the course of time. Model (6.1) is a mathematical expression of such a development. For the Markov sequence, the time interval can be broken at any point i for conditionally independent past $\Gamma^- = \{x^k : k < i\}$ and future $\Gamma^+ = \{x^k : k > i\}$. However, RF is defined on an n-dimensional domain Ω. For its geometrical partitioning into two parts Γ^- and Γ^+, at least an $(n-1)$-dimensional domain Γ is required. Markov RF suggests that for any set Γ (of a certain class) all RV included in Γ^- are conditionally independent of RV belonging to Γ^+, when values of Γ are known. It is possible to name Γ^-, Γ, and Γ^+ as past, present, and future only roughly.

However, Markov property allows us to imagine RF as one developing in time from Γ^- through Γ to Γ^+, and in addition, Γ moves along Ω with the course of time. For example, if we assume that the lines of a two-dimensional grid Ω represent Γ, the field is formed line by line.

Further development of this idea allows us to generalize autoregressive (AR) models of random sequences on RF. If the procedure for scalar sequence formation $x^0, x^1, x^2, \ldots$ usually corresponds to temporally observed values, then the procedure of field $X = \{x_{\bar{j}} : \bar{j} \in \Omega\}$ formation needs special attention. For this purpose, it is necessary to order grid Ω nodes. Then, it is possible to say about any two elements of the field that one of the elements precedes the other. If $x_{\bar{i}}$ precedes $x_{\bar{j}}$, let's mark it as $(\bar{i}) < (\bar{j})$.

6.2.4.1 *Linear autoregressive models of random fields*

The simplest autoregressive model is a linear stochastic equation

$$x_{\bar{j}} = \sum_{\bar{l} \in G_j} \alpha_{\bar{j},\bar{l}} x_{\bar{l}} + \sum_{\bar{l} \in Y_j} \beta_{\bar{j},\bar{l}} \xi_{\bar{l}}, \quad \bar{j} \in J, \tag{6.5}$$

with white Gaussian RF $\{\xi_{\bar{j}}\}$, which corresponds to a well-known autoregressive-moving average equation for random sequences. However, unlike its one-dimensional prototype, properties of RF

generated by (6.5) are not thoroughly understood, even for constant coefficient models $\alpha_{\bar{j},\bar{l}} = \alpha_{\bar{l}}$, $\beta_{\bar{j},\bar{l}} = \beta_{\bar{l}}$ and unchanging domains $G_{\bar{j}} = G$ и $Y_{\bar{j}} = Y$:

$$x_{\bar{j}} = \sum_{\bar{l}\in G} \alpha_{\bar{l}} x_{\bar{j}-\bar{l}} + \sum_{\bar{l}\in Y} \beta_{\bar{l}} \xi_{\bar{j}-\bar{l}}, \quad \bar{j} \in J. \tag{6.6}$$

An important particular case of (6.6) is represented by equations of multidimensional AR

$$x_{\bar{j}} = \sum_{\bar{l}\in G} \alpha_{\bar{l}} x_{\bar{j}-\bar{l}} + \xi_{\bar{j}}, \quad \bar{j} \in J. \tag{6.7}$$

The above-mentioned equations describe the algorithm of RF formation $\{x_{\bar{j}}\}$ at the point $\bar{j} = (j_1, j_2, \ldots, j_n)$. Besides, it is assumed that we have at our disposal all values of $x_{\bar{j}-\bar{l}}$, $\bar{l} \in G$, $(\bar{j} - \bar{l}) < (\bar{j})$, and RF calculated before or set as entry conditions.

Model (6.7) has a disadvantage which complicates its analyses. It is probably a large number of summands on the right side of the equation. However, it is possible to reduce their number to a minimum in the following way. The simplest AR equation generating n-dimensional field X, which does not fall into independent fields of less dimension has the following form:

$$x_{\bar{i}} = \sum_{k=1}^{n} \alpha_k x_{\bar{i}-\bar{e}_k} + \beta \xi_{\bar{i}}, \tag{6.8}$$

where $\bar{e}_k = (0, \ldots, 0, 1, 0, \ldots, 0)$ is a unit vector of kth coordinate axis. Any model (6.7) can be reduced to a model of type (6.8) with a minimal number of summands. For this purpose, let's use vector autoregressive models, which in their linear form are described by the following equation:

$$\bar{x}_{\bar{i}} = \sum_{\bar{j}\in D} A_{\bar{j}} \bar{x}_{\bar{i}+\bar{j}} + B\bar{\xi}_{\bar{i}}, \tag{6.9}$$

where $\bar{x}_{\bar{i}}$ is the value of a vector field in the node $\bar{i}$, $A_{\bar{j}}$, B are square matrices, and $\{\bar{\xi}_{\bar{i}}\}$ is a renewing standard vector field of independent vectors with independent components. Indeed, let us consider the

Habibi model as an example

$$x_{i_1 i_2} = \rho_1 x_{i_1-1,i_2} + \rho_2 x_{i_1,i_2-1} - \rho_1\rho_2 x_{i_1-1,i_2-1} + \sigma\sqrt{(1-\rho_1^2)(1-\rho_2^2)}\xi_{i_1 i_2}. \tag{6.10}$$

Its calculation scheme is shown in Figure 6.3(a). Let's imagine this model in the form of (6.9). For this purpose, we shall introduce vectors $\bar{x}_{\bar{i}} = \bar{x}_{ij} = (x_{ij}, x_{i,j-1})^T$ and $\bar{\xi}_{\bar{i}} = (\xi_{ij}, \varphi_{ij})^T$, $\beta = \sigma\sqrt{(1-\rho_1^2)(1-\rho_2^2)}$, then $\bar{x}_{\bar{i}-\bar{e}_1} = (x_{i-1,j}, x_{i-1,j-1})^T$, $\bar{x}_{\bar{i}-\bar{e}_2} = (x_{i,j-1}, x_{i,j-2})^T$, and (6.10) will be equal to the first component of the vector equation

$$\begin{pmatrix} x_{ij} \\ x_{i,j-1} \end{pmatrix} = \begin{pmatrix} \rho_1 & -\rho_1\rho_2 \\ 0 & 0 \end{pmatrix} \begin{pmatrix} x_{i-1,j} \\ x_{i-1,j-1} \end{pmatrix} + \begin{pmatrix} \rho_2 & 0 \\ 1 & 0 \end{pmatrix} \begin{pmatrix} x_{i,j-1} \\ x_{i,j-2} \end{pmatrix} + \begin{pmatrix} \beta & 0 \\ 0 & 0 \end{pmatrix} \begin{pmatrix} \xi_{ij} \\ \varphi_{ij} \end{pmatrix} \tag{6.11}$$

or at obvious notation

$$\bar{x}_{\bar{i}} = A_1 \bar{x}_{\bar{i}-\bar{e}_1} + A_2 \bar{x}_{\bar{i}-\bar{e}_2} + B\bar{\zeta}_{\bar{i}},$$

that is the minimal vector model of type (6.9). Figure 6.2(b) shows the elements of the field, included in the vectors of model (6.11).

First, let us analyze the main probabilistic characteristics of multidimensional AR RF models. This class of RF is generated by linear stochastic difference equations of the following form:

$$x_{\bar{i}} = \sum_{\bar{j}\in D} \alpha_{\bar{j}} x_{\bar{i}-\bar{j}} + \beta\xi_{\bar{i}}, \quad \bar{i} \in \Omega, \tag{6.12}$$

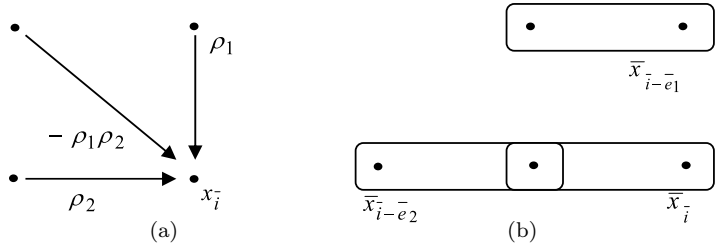


Figure 6.2. Transition from scalar model (a) to vector model (b).

where $X = \{x_{\bar{i}}, \bar{i} \in \Omega\}$ is a modeled RF defined on an N-dimensional grid $\Omega = \{\bar{i} = (i_1, i_2, \ldots, i_N) : i_k = 1 \ldots M_k,\ k = 1 \ldots N\}$, $\{\alpha_{\bar{j}}, \beta, \bar{j} \in D\}$ are model coefficients, $\Xi = \{\xi_{\bar{i}}, \bar{i} \in \Omega\}$ is a renewing white RF, and $D \subset \Omega$ is a casual field.

In its general form, the task of model analysis (6.12) was described in [17,18,36]. A linear spatial filter with transfer function corresponds to the model (6.12):

$$H(\bar{z}) = \frac{\beta}{1 - \sum_{\bar{j} \in D} \alpha_{\bar{j}} \bar{z}^{-\bar{j}}}, \tag{6.13}$$

where $\bar{z}^{-\bar{j}} = z_1^{-j_1} z_2^{-j_2} \ldots z_N^{-j_N}$. Besides, RF spectral density of X is written as

$$S_x(\bar{z}) = H(\bar{z}) S_\xi(\bar{z}) H(\bar{z}^{-1}) = \sigma_\xi^2 H(\bar{z}) H(\bar{z}^{-1}),$$

where σ_ξ^2 is the RF variance Ξ.

RF correlation function of X can be calculated by means of backward z-transformation of spectral density:

$$R(\bar{r}) = \frac{\beta^2}{(2\pi i)^N} \oint_{C_N} S_x(\bar{z}) \bar{z}^{\bar{r} - \bar{1}} d\bar{z},$$

where $C_N = \{|\bar{z}| = 1\}$ is a unit polycircle in multidimensional complex space.

The drawback of such models is the impossibility to describe isotropic RF, e.g. fields with CF $R(\bar{r}) = R(\bar{r} = \sqrt{r_1^2 + r_2^2 + \ldots r_N^2})$. However, the analysis shows that by means of trial-and-error procedure, it is possible to obtain quite acceptable approximations.

In [27–30], in order to obtain close to isotropic RF, the authors suggest using one-dimensional filters with multiple roots of characteristic equations: $1 - \sum_{j=1}^{n_k} \alpha_{kj} \lambda_k^j = 0$, where $n_k, k = 1 \ldots M$ are orders of one-dimensional AR.

In this section, the authors try to solve the synthesis problem of an N-dimensional RF model with multiple roots of characteristic equations of one-dimensional filters and to analyze its probabilistic properties. First, let us solve these tasks for a one-dimensional model.

Let's consider one-dimensional AR of M length:

$$x_i = \sum_{j=1}^{n} \alpha_j x_{i-j} + \beta \xi_i, \quad i = 1 \ldots M. \tag{6.14}$$

Here $\{\xi_i\}$ is a Gaussian sequence of independent components with mathematical expectation 0 and variance σ_ξ^2. To solve the synthesis task, it is necessary to determine the AR coefficients $\{\alpha_j,\ j =$

$1 \dots n;\ \beta\}$ using the given root of a characteristic equation $1/\rho$, its multiplicity to n, and the desired field variance σ_x^2. In the case of multiple roots, this equation can be written in an operator form as follows:

$$(1 - \rho z^{-1})^n x_i = \beta \xi_i, \quad i = 1 \dots M, \tag{6.15}$$

where z^{-1} is a shift operator. Provided that $z^{-k} x_i = x_{k-i}$, let's rewrite (6.15):

$$x_i = \sum_{j=1}^{n} (-1)^{j+1} C_n^j \rho^j x_{i-j} + \beta \xi_i, \quad i = 1 \dots M. \tag{6.16}$$

Comparison of (6.15) and (6.16) allows us to write the expression for coefficients $\alpha_j = \alpha_j(\rho, n)$ as

$$\alpha_j(\rho, n) = (-1)^{j+1} C_n^j \rho^j, \quad j = 1 \dots n. \tag{6.17}$$

The value of the unknown parameter β, which is an amplification coefficient in transfer function (6.13), should be selected so as to make the filter stable. Further, it will be shown how to determine β on the basis of a CF model.

One of the tasks of model statistical analysis is to obtain its CF:

$$R(k) = \beta^2(\rho, n) \rho^k \sum_{\ell=0}^{n-1} g(n, \ell, k) \frac{\rho^{2(n-\ell-1)}}{(1 - \rho^2)^{2n-\ell-1}}, \tag{6.18}$$

where

$$g(n, \ell, k) = \frac{(n + k - 1)!(2n - \ell - 2)!}{\ell!(n - 1)!(n - \ell - 1)!(n + k - \ell - 1)!} \tag{6.19}$$

and coefficient $\beta = \beta(\rho, n)$ can be obtained from equation $R(0) = 1$:

$$\beta^2(\rho, n) = \frac{(1 - \rho^2)^{2n-1}}{\sum_{\ell=0}^{n-1} (C_{n-1}^{\ell} \rho^{\ell})^2}. \tag{6.20}$$

Formulae (6.18–6.20) give a general view of a normalized autocorrelation function (NAF) of a one-dimensional model (6.14) for given ρ and n. In order to obtain CF at variances σ_x^2 and σ_ξ^2, which are not equal to 1, it is necessary to multiply the right-hand side of equation

(6.20) by σ_x^2/σ_ξ^2. Thus, we obtain the expression for coefficient β as well:

$$\beta = \frac{\sigma_x}{\sigma_\xi}\sqrt{\frac{(1-\rho^2)^{2n-1}}{\sum_{\ell=0}^{n-1}\left(C_{n-1}^{\ell}\rho^{\ell}\right)^2}}. \tag{6.21}$$

Formulae (6.17) and (6.20) completely determine the unknown coefficients of one-dimensional AR model (6.14) with multiple roots of the characteristic equation.

Now let's consider an N-dimensional case. RF model for a given variance σ_x^2 is completely determined by a parameter vector $(\rho_1, \rho_2, \ldots, \rho_N)$ and a multiplicity vector $(n_1, n_2, \ldots, n_N)$.

Figure 6.3 is an image frame (600×400 elements) obtained on the basis of the model with multiple roots. Output analysis shows that by changing coupling parameters and the multiplicity formula, it is possible to obtain a wide range of different patterns, which can help to develop complex models of multizone images.

At the same time if root multiplicity grows the modeled RF approaches in its properties to the isotropic RF. This is also confirmed by cross section shape of equal level shown in Figure 6.4.

The advantages of the suggested methodology for assessing RF isotropy are the relative simplicity of calculation and the possibility of using it on different samples of real data.

6.2.5 *Wave models of random fields*

Let's consider an RF wave model, which generalizes a number of other models and helps solve the tasks of correlation analysis and synthesis [18,20,43] effectively. This model is simple enough and can serve as a basis for simulating images and their sequences with a given CF without increasing the number of model parameters.

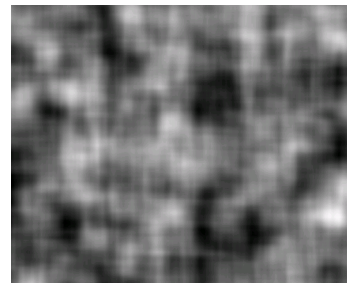
In a wave model, RF is determined by the equation

$$x_{\bar{j}}^{t} = \sum_{\{k:\tau_k \le t\}} f((\bar{j}, t), (\bar{u}_k, \tau_k), \bar{\omega}_k), \tag{6.22}$$

where an $(n+1)$-dimensional domain $\{(\bar{j}, t)\}$ may be net or continuous, $\{(\bar{u}_k, \tau_k)\}$ is a discrete field of random points (FRP) in an $(n+1)$-dimensional continuous space, t and τ_k are interpreted as time, and $\bar{\omega}_k$ is a random vector of function f parameters.

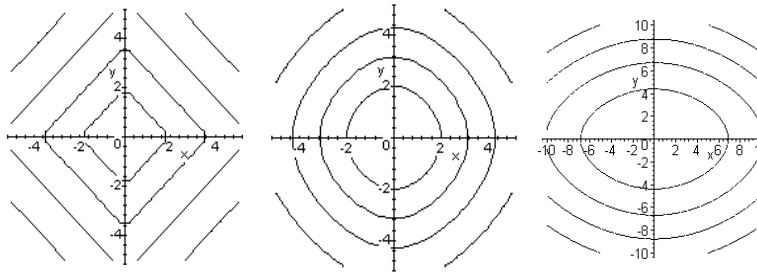


(a)



(b)

Figure 6.3. Representation of random fields on the basis of autoregression with multiple roots. (a) multiplicity (2, 2), parameter vector (0.95, 0.95); (b) multiplicity (3, 3), parameter vector (0.95, 0.95).



(a) (b) (c)

Figure 6.4. Cross-sections of a correlation function for a two-dimensional model with square roots. (a) multiplicity (1, 1), parameters (0.9, 0.9); (b) multiplicity (3, 3), parameters (0.9, 0.9); (c) multiplicity (2, 3), parameters (0.9, 0.9).

This field can be represented as the effect of random disturbances or waves $f((\bar{j},t),(\bar{u}_k,\tau_k),\bar{\omega}_k)$, generated in random places $\bar{u}_k$

at random time τ_k and changing according to a given law in time and space.

Selection of function f, FRP parameters, and $\bar{\omega}$ allows us to obtain a vast class of fields, which includes the following models:

1. *Poisson fields*: Where $f((\bar{j},t),(\bar{u}_k,\tau_k),\bar{\omega}_k) = \delta((\bar{j},t)-(\bar{u}_k,\tau_k))$, and δ is a Kronecker delta and $\{(\bar{u}_k,\tau_k)\}$ is Poisson FRP.
2. *Multidimensional filtered Poisson process*: When $f((\bar{j},t),(\bar{u}_k,\tau_k),\bar{\omega}_k) = g((\bar{j}-\bar{u}_k t-\tau_k),\xi_k)$, where $\{\xi_k\}$ is a system of scalar RF. This model can generate only stationary homogeneous fields, and the generating waves may differ from each other only in one parameter ξ_k.
3. *Weighted sum model*: It is obtained at waves $f((\bar{j},t),(\bar{u}_k,\tau_k),\bar{\omega}_k) = g((\bar{j},t),(\bar{u}_k,\tau_k))\xi_k$, where $\{(\bar{u}_k,\tau_k)\}$ is a set of all grid nodes and g represents corresponding weights of random variables ξ_k. This model corresponds to different RF expansions into a system of basic functions, e.g. [37].
4. *Random walk model*: FRP describes the random walk (perhaps with appearance and disappearance) of a set of waves, and the choice $\bar{\omega}_k$ determines the dynamics of a wave shape and intensity. Among other things, such models can be applied to simulate moving clouds.

Let's consider a particular case of a wave model for which correlation tasks of analysis and synthesis can easily be solved. Suppose

$$f((\bar{j},t),(\bar{u}_k,\tau_k),\bar{\omega}_k) = g(\rho_k/R_k)\exp(-\mu/|t-\tau_k|)\xi_k, \qquad (6.23)$$

where FRP is a Poisson one with constant density λ, $\rho_k = |\bar{j}-\bar{u}_k|$ is a distance between $\bar{j}$ and $\bar{u}_k$, $\{R_k\}$ is a system of independent non-negative equally distributed RF with PDF $w(\alpha)$, and $\{\xi_k\}$ is a system of independent equally distributed RF. Waves are motionless and independent, have spherical sections in space, and exponentially attenuate over time; system $\{\xi_k\}$ determines wave intensity, and $\{R_k\}$ is their spatial scale. Evidently, generated field X is stationary and homogeneous and has zero mean and isotropic spatial CF.

Provided that in this case all summands in (6.22) are non-correlated and elementary event ΔA ={in the element $\Delta V = \Delta_{j_1}\Delta_{j_2}\ldots\Delta_{j_n}\Delta\tau$ there appeared a point in FRP which corresponds to a wave with spatial scale α from the element $\Delta\alpha$} has probability $P(\Delta A) = \lambda\Delta V w(\alpha)\Delta\alpha$, let's express (6.23) by an integral over

variables τ, α, $j_1, \ldots, j_n$. After integration over τ and taking into account $g(y) = c\exp(-2y^2)$, we obtain

$$V(\rho, t) = \frac{c^2\pi^{n/2}\lambda}{2^{n+1}} e^{-\mu t} \int_0^\infty \alpha^n \exp\left(-\frac{\rho^2}{\alpha^2}\right) w(\alpha) d\alpha. \tag{6.24}$$

When $\rho = t = 0$, let's find field variance from (6.24)

$$\sigma_n^2 = \frac{c^2\pi^{n/2}\lambda}{2^{n+1}\mu} M[R^n].$$

It is proportional to FRP density λ, efficiency interval $1/\mu$ of wave attenuation, and mean value of the nth degree of scale R.

Simulation of a discrete field on an n-dimensional grid $\{\bar{j}\}$ with sample time Δt on the base of a particular model can be implemented with the help of the following algorithm. At the initial time $t_0 = 0$, field values in all nodes are equal to zero. At each subsequent moment $t_m = m\Delta t$, Poisson FRP with density $\lambda\Delta t$ is formed over continuous space or grid, which somehow overlaps $\{\bar{j}\}$. At each generated point of FRP $\bar{u}_k$ RV ξ_k and R_k are formed. After that, the following transformation of all field values on grid $\{\bar{j}\}$ is carried out

$$x_{\bar{j}}^{t_m} = x_{\bar{j}}^{t_{m-1}} \exp(-\mu \cdot \Delta t) + \sum_k g(\rho_k/R_k)\xi_k. \tag{6.25}$$

In this simulation, only large summands (in comparison with quantization level) can be taken into account. The advantage of this algorithm is its recurrence, as it makes it easy to implement field simulation on a computer.

Figure 6.5 represents an example of image simulation with the described wave model. It is necessary to underline that procedure (6.25) actually implements a time-varying image. Therefore, this figure shows only one frame of this process.

Each field value $x_{\bar{j}}^t$ is a sum of random numbers of RV, so generally speaking, the field will not be Gaussian even with Gaussian $\{\xi_k\}$. However, when the model parameters $\lambda M[R^n]/\mu$ grow, then in (6.25), the number of summands with similar distributions increases and the field is normalized.

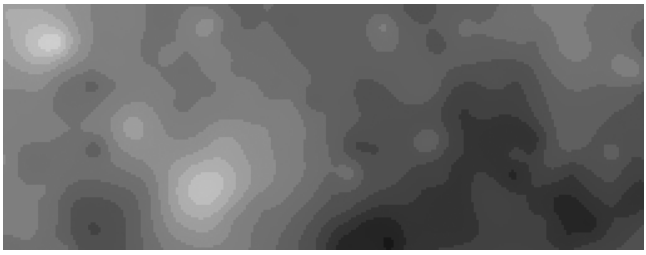


Figure 6.5. Example of image implementation using a wave model.

Let us now consider the solution of correlation analysis and synthesis task. From (6.24), it follows that the formed field has an exponential time NAF $e^{-\mu t}$ and space NAF

$$r(\rho) = \frac{1}{M[R^n]} \int_0^\infty \alpha^n \exp(-\rho^2/\alpha^2)\omega(\alpha)d\alpha. \tag{6.26}$$

Thus, solving analysis tasks when PDF $\omega(\alpha)$ is set, the required NAF can be found analytically or by numerical integration. Solving synthesis tasks when NAF $r(\rho)$ is given, it is necessary to solve integral equation (6.26) for unknown $\omega(\alpha)$.

As it is not always possible to find an analytical solution to synthesis task, let's consider a method of its approximate solution. From (6.26), it follows that at degenerate distribution ($R = \alpha = \text{const}$) we obtain CF $\exp(-\rho^2/\alpha^2)$.

6.2.6 *Doubly stochastic models of images*

In the literature, doubly stochastic models (DSM) of RFs are known. For instance, in [9], the authors propose using combinations of different methods for forming RFs, for example,

$$B(x) = \eta(x)B_1(x) + (1 - \eta(x))B_2(x). \tag{6.27}$$

where $B_1(x)$ is the implementation of one microstructure (for example, an object on the image), $B_2(x)$ is the implementation of another microstructure (for example, the underlying surface), and $\eta(x)$ is the implementation of the RF that defines the interaction between these microstructures. In this case, the multipliers $\eta(x)$ and $(1-\eta(x))$ provide switching from one implementation $B_1(x)$ to the other $B_2(x)$. However, the solutions obtained either do not allow the description

of the variety of characteristics of real images or are associated with considerable computational difficulties. To overcome these difficulties, we will assume that an RF is defined on a rectangular multidimensional grid Ω so that its values are $x_{\bar{i}} = F(x_{\bar{j}}, \alpha_{\bar{i}}, \xi_{\bar{i}})$, where $\bar{i} \in \Omega$, $F()$ is some transformation, and $\alpha_{\bar{i}}$ are parameters that represent an RF independent of $\xi_{\bar{i}}$. Then the distribution density of a random variable can be defined as $\omega(x_{\bar{i}}) = \int_D \omega(x_{\bar{i}}|\alpha_{\bar{i}})\omega(\alpha_{\bar{i}})d\alpha_{\bar{i}}$. The coefficients $\alpha_{\bar{i}}$ characterize the size and the shape of objects on the image to be modeled; an increase in the mean values of these parameters leads to an increase in the mean size of objects on the image modeled. The choice of the method of transformation of brightness values into a set of correlation parameters allows one to control the values of the correlation functions of the image, which makes it possible to model RFs whose correlation properties are close to those of real satellite images.

As an example, consider the implementation of a two-dimensional RF $X = \{x, \bar{i} \in \Omega\}$ obtained by a model with multiple roots of the characteristic equation [43]. In this case, simulated images are generated by one-dimensional autoregressions [42, 43]. One can find models of two-dimensional RFs for one-dimensional autoregressions of different orders. For example, for a first-order AR process $x_i = 2\rho x_{i-1} + \rho^2 x_{i-2} + \xi_i$, one can obtain the corresponding model of a two-dimensional RF:

$$\begin{aligned} x_{ij} = {} & 2\rho_y x_{i-1,j} + 2\rho_x x_{i,j-1} - 4\rho_y\rho_x x_{i-1,j-1} \\ & - \rho_y^2 x_{i-2,j} - \rho_x^2 x_{i,j-2} + 2\rho_y^2\rho_x x_{i-2,j-1} \\ & + 2\rho_x^2\rho_y x_{i-1,j-2} - \rho_y^2\rho_x^2 x_{i-2,j-2} + b\xi_{ij}. \end{aligned} \tag{6.28}$$

Suppose that the parameters ρ_x and ρ_y used in this model are realizations of two basic RFs the values of brightness of one of which are transformed into a set of correlation parameters $\{\rho_{xij}, i = 1, \ldots, M1, j = 1, \ldots, M2\}$, while those of the other, into a set of correlation parameters $\{\rho_{yij}, i = 1, \ldots, M1, j = 1, \ldots, M2\}$ according to the following relations:

$$\rho_{xij} = r_{1x}\rho_{x(i-1)j} + r_{2x}\rho_{xi(j-1)} - r_{1x}r_{2x}\rho_{x(i-1)(j-1)} + \varsigma_{xij} \quad \text{and}$$

$$\rho_{yij} = r_{1y}\rho_{y(i-1)j} + r_{2y}\rho_{yi(j-1)} - r_{1y}r_{2y}\rho_{y(i-1)(j-1)} + \varsigma_{yij},$$

where $\{\varsigma_{xij}\}$ and $\{\varsigma_{yij}\}$ are two-dimensional RFs of independent Gaussian random variables with zero means and variances $M(\varsigma_{xij}^2) =$

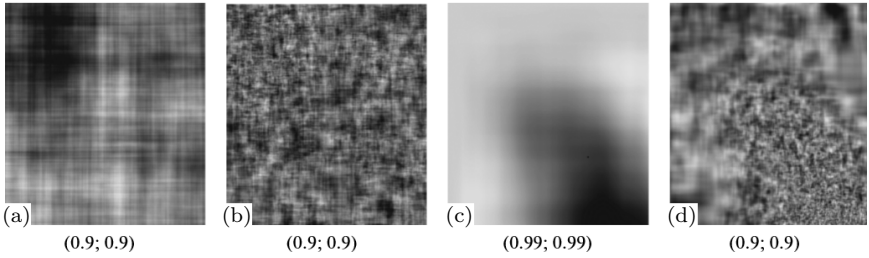


Figure 6.6. Simulation of images with variable correlation properties.

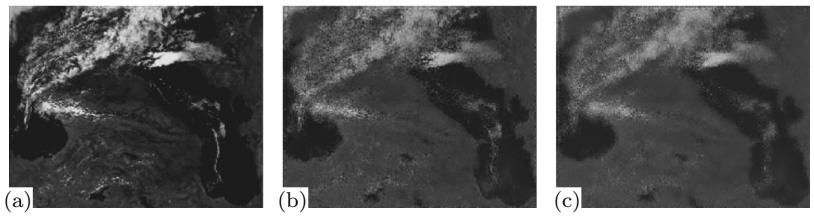
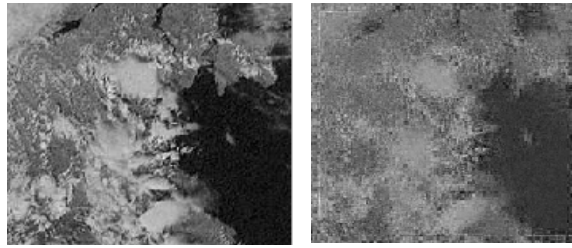


Figure 6.7. Formation of an image with variable correlation properties.

$(1 - r_{1x}^2)(1 - r_{2x}^2)\sigma_{\rho_x}^2$ and $M(\varsigma_{yij}^2) = (1 - r_{1y}^2)(1 - r_{2y}^2)\sigma_{\rho_y}^2$; $\sigma_{\rho_x}^2 = M(\rho_{xij}^2)$, $\sigma_{\rho_y}^2 = M(\rho_{yij}^2)$. The choice of the method of transformation of brightness values into a set of correlation parameters allows one to control the values of the correlation function of the image [13], which makes it possible to simulate RFs with correlation properties close to those of real satellite images.

An example of simulation of an image with variable correlation properties that characterize the relief of the assumed image of the earth's surface is shown in Figure 6.6, where Figures 6.6(a) and 6.6(c) represent the basic RF and Figures 6.6(b) and 6.6(d) represent a frame of the image with variable correlation properties. To form RFs in Figures 6.6(a) and 6.6(b), we used the Habibi model, and to form RFs in Figures 6.6(c) and 6.6(d), we used an RF model with multiple roots of multiplicity (2; 2).

Figure 6.7 shows an example of using the method of simulation of a real image: Figure 6.7(a) is a real image of size 440×440, Figure 6.7(b) is a DSM of multiplicity (2; 2) of the image, and Figure 6.7(c) is Habibi's DSM of the image. The statistical estimation of the parameters of the image was performed in a sliding window of size 15×15.



(a) (b)

Figure 6.8. Source image (a); model realization (b).

The solution of identifying parameters task of two-dimensional model (6.28) deals with the estimation of the statistical parameters $m_{\rho x}$, $m_{\rho y}$, $\sigma^2_{\rho x}$, and $\sigma^2_{\rho y}$. Furthermore, it is necessary to estimate the constant parameters r_{1x}, r_{2x}, r_{1y}, r_{2y}. It is quite easy to use the sliding window method for such estimation.

Figure 6.8 shows how to represent source image (a) by slide window estimation using the model with varying parameters (b).

The other way to estimate DSM parameters is the use of pseudo-gradient (PG) search algorithms [30]. PG algorithms provide the accuracy of estimation, which is sufficient for most practical applications, without requiring an *a priori* knowledge of the gradient of the objective function and features of its behavior. Consider, for example, the use of PG algorithms for identifying the parameters of a two-dimensional DSM model. Let us write down the conditions for choosing the direction of the PG search in the following form:

$$\{\tilde{\rho}_{xij}\} = \min_{(\hat{B}_1 - B_{01})^2 + (\hat{B}_2 - B_{02})^2} (a_x \quad b_x \quad c_x \quad d_x \quad e_x \quad f_x \quad g_x \quad h_x \quad k_x),$$

$$\{\tilde{\rho}_{yij}\} = \min_{(\hat{B}_1 - B_{01})^2 + (\hat{B}_2 - B_{02})^2} (a_y \quad b_y \quad c_y \quad d_y \quad e_y \quad f_y \quad g_y \quad h_y \quad k_y),$$

where the following formula describes the direction of correlation parameters decreasing

$$\begin{aligned}(a_x) = {} & (r_{1xs} - \Delta \mathrm{r}_{1xs})\tilde{\rho}_{xi-1,j} + (r_{2xs} - \Delta \mathrm{r}_{2xs})\tilde{\rho}_{xi,j-1} - (r_{1xs} - \Delta \mathrm{r}_{1xs}) \\ & \times (r_{2xs} - \Delta \mathrm{r}_{2xs})\tilde{\rho}_{xi-1,j-1} \\ & + \sigma_x \sqrt{(1 - (r_{1xs} - \Delta \mathrm{r}_{1xs})^2}\sqrt{(1 - (r_{2xs} - \Delta \mathrm{r}_{2xs})^2}\varsigma_{ij}.\end{aligned}$$

The following formula describes the direction of the row correlation parameter decreasing and not changing column correlation

parameter:

$$(b_x) = (r_{1xs} - \Delta r_{1xs})\tilde{\rho}_{xi-1,j} + r_{2xs}\tilde{\rho}_{xi,j-1} - (r_{1xs} - \Delta r_{1xs}) \times r_{2xs}\tilde{\rho}_{xi-1,j-1} + \sigma_x\sqrt{(1-(r_{1xs}-\Delta r_{1xs})^2}\sqrt{(1-r_{2xs}^2}\varsigma_{ij}.$$

The following formula describes the direction of row correlation parameter decreasing and column correlation parameter increasing:

$$(c_x) = (r_{1xs} - \Delta r_{1xs})\tilde{\rho}_{xi-1,j} + r_{2xs}\tilde{\rho}_{xi,j-1} - (r_{1xs} - \Delta r_{1xs}) \times r_{2xs}\tilde{\rho}_{xi-1,j-1} + \sigma_x\sqrt{(1-(r_{1xs}-\Delta r_{1xs})^2}\sqrt{(1-r_{2xs}^2}\varsigma_{ij}.$$

Similarly, you can specify the other ways to change correlation parameters. In this case, RF is implemented for each pair of parameters $\tilde{\rho}_{xij}$ and $\tilde{\rho}_{yij}$.

When the calculated statistics is similar to that obtained from the observed image, the values r_{1xs}, r_{2xs}, r_{1ys}, r_{2ys} are considered as the results of identifying.

Analysis of the obtained results shows that PG algorithm estimates are much more accurate than estimates which are got in the sliding window (10–15 times).

Thus, the method considered for forming an image with variable properties allows one to model images that are close to real ones in their statistical properties.

6.3 Image Filtering

Extraction of useful image components based on their noise observations is an important task since it gives the opportunity to improve the multidimensional image on the background of noise. Furthermore, as is shown in the following section, anomaly detection problem also makes it necessary to determine covariance matrices of filtering errors. However, there exist many different approaches to solve the problem of determining the potential accuracy of estimation and construction of multidimensional RF filtering algorithms. In the first case, in order to obtain relatively simple analytical relations, it is advisable to refer to asymptotic formulae of the Wiener filter. While constructing multidimensional image estimation on the background noise, it is desirable to find optimal algorithm structures (or structures close to optimal) with low computational cost. It can be done

using recursive estimation methods based on the above-mentioned image models.

6.3.1 *Optimal image filtering efficacy*

To find optimal filtering error variance σ_ε^2 let's consider homogeneous information RF $x_{\bar{j}}$, set on an infinite n-dimensional grid J. Suppose that using observations $z_{\bar{j}} = x_{\bar{j}} + \theta_{\bar{j}}$ it is necessary to give the best (error variance minimum) linear estimation $\widehat{x}_{\bar{0}} = \sum_{\bar{j}\in J} h_{\bar{j}} z_{\bar{j}}$ of information RF element $x_{\bar{0}}$. Condition for minimum σ_ε^2 can be written as a system of linear equations:

$$h_{\bar{q}}\sigma_\theta^2 + \sum_{\bar{j}\in J} h_{\bar{j}} R(\bar{q} - \bar{j}) = R(\bar{q}), \quad \bar{q} \in J, \tag{6.29}$$

which can be considered as an n-dimensional analog of the Wiener–Hopf equations; $R(\bar{q}) = M\{x_{\bar{j}}, x_{\bar{j}+\bar{q}}\}$ is a covariation function. In addition, the smallest error variance is $\sigma_\varepsilon^2 = \sigma_\theta^2 h_{\bar{0}}$. Unfortunately, the analytical solution (6.28) can be found only for a very small class of "separable" exponential correlation functions $R(\bar{q}) = \sigma_x^2 \prod_{i=1}^n \rho_i^{q_i}$ [7]. However, supposing that the spatial grid J cells are small in comparison with RF, it is possible to replace system (6.29) with one integral equation. Then

$$\sigma_\varepsilon^2 = \sigma_\theta^2 h_{\bar{0}} \cong \sigma_\theta^2 \frac{1}{(2\pi)^n} \int_{-\infty}^{\infty} f(\bar{\lambda}) d\bar{\lambda} / (\sigma_\theta^2 + f(\bar{\lambda})), \tag{6.30}$$

where

$$f(\bar{\lambda}) = \int_{R^n} R(\bar{u}) \exp(-j(\bar{\lambda}, \bar{u})) d\bar{u} \tag{6.31}$$

is RF spectral density $x(\bar{u}), \bar{u}, \bar{\lambda} \in R^n$; $(\bar{\lambda}, \bar{u}) = \sum_{k=1}^n \lambda_k u_k$; $d\bar{\lambda} = d\lambda_1 d\lambda_2 \ldots d\lambda_n$.

Thus, to analyze filtering efficiency, it is sufficient to find spectral density (6.31) of information RF and to make calculations using (6.30). The greatest challenge is usually connected with n-multiple integration of (6.30) and, particularly, (6.29). These problems may be significantly lower when RF $x(\bar{u})$ is isotropic throughout space R^n or any other subspaces $\Omega_m \subset R^n$. Indeed, after entering spherical

coordinates in R^n (6.30) and (6.31) for isotropic RF can be written in the following form:

$$\frac{\sigma_\varepsilon^2}{\sigma_\theta^2} = \frac{1}{(2\pi)^{n-1}\Gamma(0.5n)} \int_0^\infty \frac{k^{n-1} f(k)}{\sigma_\theta^2 + f(k)} dk, \tag{6.32}$$

$$f(k) = (2\pi)^{n/2} \int_0^\infty R(\rho)\rho^{n-1} \frac{J_{0.5n-1}}{(k\rho)^{0.5n-1}} d\rho, \tag{6.33}$$

where $J_\nu(\cdot)$ are Bessel functions of ν order, $\Gamma(\cdot)$ is a complete gamma function, $k = |\bar{\lambda}|$, and $\rho = |\bar{u}|$.

For isotropic RF on spaces with odd number dimensions, Bessel functions in (6.32) can be expressed by means of elementary functions. In these cases, calculations for (6.32) and (6.33) are rather easy. For example, for isotropic RF with

$$R(\rho) = \sigma_x^2 \exp(-\alpha\rho), \quad \rho = \sqrt{\sum_{k=1}^{n} u_k^2},$$

we obtain

$$\frac{\sigma_\varepsilon^2}{\sigma_\theta^2} = \frac{\gamma q}{(2\pi)^{n-1}\Gamma(0.5n)} \int_0^\infty \frac{\nu^{n-1}\Phi_n(\nu)}{\gamma + \Phi_n(\nu)} d\nu, \tag{6.34}$$

where $\gamma = \alpha^n/q$; $q = \sigma_x^2/\sigma_\theta^2$ is information and interference RF variance ratio, $\Phi_1(\nu) = 2/(1+\nu^2)$, $\Phi_3(\nu) = 8\pi/(1+\nu^2)^2$; $\Phi_5(\nu) = 64\pi^2/(1+\nu^2)^3$; $\Phi_7(\nu) = 96\pi^3/(1+\nu^2)^4$, etc.

For isotropic exponentially correlated RF, set on spaces with even number of dimensions $n = 2N$, there is a simple formula for integral 34:

$$f(k) = (2\pi)^M \sigma_x^2 k^{-2(M-1)} (-1)^M \frac{d^M}{d\alpha^M} \left(\frac{(\sqrt{\alpha^2+k^2} - \alpha)^{M-1}}{\sqrt{\alpha^2+k^2}} \right).$$

Unfortunately, for large dimensions n of homogeneous RF, rather complicated formulae can be obtained, and it is advisable to use numerical methods while calculating in (6.33). At the same time, according to the analysis, if the generalized parameter $\gamma = \alpha^n/q$

increases, then the variance of optimal filtering error of considered isotropic RF converges to

$$\sigma_\varepsilon^2/\sigma_x^2 = \beta_n \sqrt[n+1]{\gamma},$$

where $\beta_1 \cong 0.707$; $\beta_2 \cong 0.66$; $\beta_3 \cong 0.63$; $\beta_4 \cong 0.606$; ... $\beta_\infty \cong 0.58$.

Thus, the considered relations give a possibility to obtain rather simple estimations of potential RF filtering accuracy on the background of noise. In addition, the obtained limit values of filtering variance allow us to construct characteristics of optimal algorithms to detect spatial anomalies on the background of multidimensional interfering images.

6.3.2 *Tensor Kalman filter*

One of the least overloaded restrictions allowing us to find a recurrent solution to the problem is a description of interfering image frame sequence by means of nonlinear tensor stochastic differential (6.1). Based on observations of frame sequence

$$z_{\bar{j}}^t = x_{\bar{j}}^t + \theta_{\bar{j}}^t, \quad \bar{j} \in J_t, \quad t = 1, 2, \ldots,$$

representing an additive mixture of information RF $x_{\bar{l}}^t$ and white RF of noise $\theta_{\bar{l}}^t$, it is necessary to find the best estimate $x_{\bar{l}}^t$ of a recurrent frame of information RF. To find such an estimation, let's use the criterion of average maximal gain [43] and an invariant imbedding method. As a result, we can obtain the following RF recursive estimation rule:

$$\begin{aligned} &\widehat{x}_{\bar{j}}^t = \widehat{x}_{\ni\bar{j}}^t + P_{\bar{j}\bar{l}}^t V_\theta^{-1}(z_{\bar{l}}^t - \widehat{x}_{\ni\bar{l}}^t), \quad \bar{l}, \bar{j} \in J_t, \\ &P_{\bar{j}\bar{l}}^t = P_{\ni\bar{j}\bar{l}}^t (E + V_\theta^{-1} P_{\ni\bar{j}\bar{l}}^t)^{-1}, \end{aligned} \tag{6.35}$$

where $\widehat{x}_{\ni\bar{j}}^t = \varphi_{\bar{j}}^t(\widehat{x}_{\bar{l}}^{t-1})$, $\bar{j} \in J_{t-1}$ is an optimal RF prediction; $P_\ni^t = (\varphi^t(\widehat{x}_{\bar{l}}^{t-1}))' P^{t-1} (\varphi^t(\widehat{x}_{\bar{l}}^{t-1}))' + \upsilon^t(\widehat{x}_{\ni\bar{l}}^t)\upsilon^t(\widehat{x}_{\ni\bar{l}}^t)$ is a covariance matrix of prediction errors; $x_{\ni\bar{j}}^1 = 0$; $P_{\ni\bar{j}\bar{l}}^1 = M\{x_{\bar{j}}^1 x_{\bar{l}}^1\}$.

Algorithm 6.35 allows us to find extrapolated RF estimates $\widehat{x}_{\ni\bar{j}}^t$ and covariance matrices of extrapolation errors $P_{\ni\bar{j}\bar{l}}^t$ recursively as

new observations $\{z^i_{\bar{j}},\ \bar{j} \in J\}$ of successive RF frames become available. For Gaussian RF defined by linear stochastic equations when $\varphi^t_{\bar{j}}(x^{t-1}_{\bar{l}}) = P^t_{\bar{j}\bar{l}} x^{t-1}_{\bar{l}}$ and $\upsilon^t_{\bar{j}\bar{q}}(x^{t-1}_{\bar{l}}) = \upsilon^t_{\bar{j}\bar{q}}(x^{t-1}_{\bar{l}})$, the procedure of filtration interpolation together with Algorithm 36 gives a strictly optimal solution on how to detect multidimensional signals on image sequence. In this case, domain G coincides with J_t in (6.35) determines the sequence of inter-frame observation processing, and while generating the log-likelihood ratio weighted summation of observations $z^t_{\bar{j}},\ \bar{j} \in J_t$ and prediction $\widehat{x}^t_{\ni\bar{j}}, \bar{j} \in J_t$ of a successive frame of a multidimensional image is used.

Tensor P^t elements are filtering error covariance x^k, which in this case depend only on four parameters, correlation coefficients r_1, r_2, and ρ, and also signal/noise ratio $q = \sigma^2_x / V_\theta$. When $q \gg 1$ and $t \to \infty$, coefficients $P^t_{j_1 j_2}$ converge to limit $\{P^t_{j_1 j_2}\} = P_{j_1 j_2}$ rather quickly. So it is often possible to apply limit values at once. It will worsen the filtering results only at the start, but it will also significantly reduce the amount of computation (or the storage space, if coefficients $\{P^t_{j_1 j_2}\}$ are calculated beforehand).

6.3.3 *Doubly stochastic nonlinear filter*

Let us use the approaches found to solve the problem of estimating the parameters of a DSM on the multidimensional grids.

The two-dimensional image can be described by RF$\{x_{ij}\}$ on the grid $\{(i,j);\ i = 1, M_1;\ j = 1, M_2\}$. Consider the doubly stochastic model of the following form:

$$x_{ij} = \rho_{xij-1} x_{i-1,j} + \rho_{yi-1j} x_{i,j-1} - \rho_{xi-1j} \rho_{yij-1} x_{i-1,j-1} + \xi_{ij}, \quad (6.36)$$

where x_{ij} is simulated RF with $M\{x_{ij}\} = 0$, $M\{x^2_{ij}\} = \sigma^2_x$, ξ_{ij} is RF of independent normal random variables with $M\{\xi_{ij}\} = 0$, $M\{\xi^2_{ij}\} = \sigma^2_{\xi ij} = \sqrt{\sigma^2_x (1 - \rho^2_{xij})(1 - \rho^2_{yij})}$, and ρ_{xij} and ρ_{yij} are varying correlation parameters by the row and column.

So we have observations $z_{ij} = x_{ij} + n_{ij}$ as a sum of information RF and white Gaussian noise $\{n_{ij}\}$ having $\sigma^2_n = M\{n^2_{ij}\}$.

For this case, we perform the construction of the best estimate minimizing the dispersion of the estimation of the filtration algorithm. To do this, we formulate the following vector $\bar{x}_{ij} =$

$(\bar{x}_{xij} \quad \bar{\rho}_{xij} \quad \bar{\rho}_{yij})^T$ with length $3(M_1+1)$, where

$$\bar{x}_{xij} = \begin{pmatrix} x_{ij} & x_{ij-1} & \cdots & x_{i1} & x_{i-1M_1} & \cdots & x_{i-1j} \end{pmatrix}^T,$$
$$\bar{\rho}_{xij} = (\rho_{xij} \quad \rho_{xij-1} \quad \cdots \quad \rho_{xi1} \quad \rho_{xi-1M_1} \quad \cdots \quad \rho_{xi-1j})^T,$$
$$\bar{\rho}_{yij} = (\rho_{yij} \quad \rho_{yij-1} \quad \cdots \quad \rho_{yi1} \quad \rho_{yi-1M_1} \quad \cdots \quad \rho_{yi-1j})^T.$$

So model (6.36) will be written as $\bar{x}_{ij} = \wp_{ij}\bar{x}_{ij-1} + \bar{\xi}_{ij}$, where

$$\bar{\xi}_{ij} = (\xi_{ij}0\ldots0)^T; \quad \wp_{ij} = \begin{pmatrix} \wp_{ijx} & 0 & 0 \\ 0 & \wp_{ij\rho x} & 0 \\ 0 & 0 & \wp_{ij\rho y} \end{pmatrix};$$

$$\wp_{ijx} = \begin{pmatrix} \rho_{xij-1} & \cdots & \rho_{yi-1j} & -\rho_{xi-1j}\rho_{yij-1} \\ 1 & \cdots & 0 & 0 \\ \cdots & \cdots & \cdots & \cdots \\ 0 & \cdots & 1 & 0 \end{pmatrix};$$

$$\wp_{ij\rho x} = \begin{pmatrix} r_{x1} & \cdots & r_{x2} & -r_{x1}r_{x2} \\ 1 & \cdots & 0 & 0 \\ \cdots & \cdots & \cdots & \cdots \\ 0 & \cdots & 1 & 0 \end{pmatrix};$$

$$\wp_{ij\rho y} = \begin{pmatrix} r_{y1} & \cdots & r_{y2} & -r_{y1}r_{y2} \\ 1 & \cdots & 0 & 0 \\ \cdots & \cdots & \cdots & \cdots \\ 0 & \cdots & 1 & 0 \end{pmatrix}.$$

The representation can be written in the general form:

$$\bar{x}_{ij} = \varphi(\bar{x}_{ij-1}) + \bar{\xi}_{ij},$$

where φ is the nonlinear matrix transformation. We introduce the extrapolated estimate $\bar{x}_{\ominus ij} = \varphi(\bar{x}_{ij-1})$ and find a matrix $\varphi'(\bar{x}_{ij-1})$. Calculations show that it will be identical to the matrix $\wp_{ij}$, except

for the first line: $\varphi_1' = (A_1\ A_2\ A_3)$, where

$$A_1 = (\rho_{xij-1} \cdots \rho_{yi-1j} - \rho_{xij-1}\rho_{yi-1j}),$$
$$A_2 = (x_{i-1j} - x_{i-1j-1}\rho_{yi-1j} \cdots 0 \quad 0),$$
$$A_3 = (0 \cdots x_{ij-1} - x_{i-1j-1}\rho_{xij-1} \quad 0).$$

Covariance matrix of extrapolation errors

$$P_{\ni ij} = M\{(\bar{x}_{\ni ij} - \bar{x}_{ij})(\bar{x}_{\ni ij} - \bar{x}_{ij})^T\}$$
$$= \varphi'(\hat{\bar{x}}_{ij-1})P_{ij-1}\varphi'(\hat{\bar{x}}_{ij-1})^T + V_{\xi ij}.$$

Using these relations and the method of recurrent vector filtration [30], we can write the following two-dimensional nonlinear filter:

$$\hat{\bar{x}}_{ij} = \hat{\bar{x}}_{\ni ij} + B_{ij}(z_{ij} - \hat{x}_{\ni ij}), \tag{6.37}$$

where $\hat{x}_{\ni ij}$ is the first element of $\hat{\bar{x}}_{\ni ij}$, $B_{ij} = P_{\ni ij}C^T D_{ij}^{-1}$, $C = (1, 0, 0, \ldots, 0)$, and $D_{ij} = CP_{\ni ij}C^T + \sigma_n^2$. Dispersion of the filtering error at each step $P_{ij} = (E - B_{ij}C)P_{\ni ij}$. The algorithm obtained has important features. First, despite the external cumbersomeness, it does not require the inversion of large-sized matrices, as in the case of line Kalman estimation [30]. Second, when forming an estimate at a point $\hat{x}_{ij}$ we use all elements on the left and on top of $\hat{x}_{ij}$. This provides reducing the estimation error. Third, the result of filtering is not only a set of estimates for $\hat{x}_{ij}$ but also the estimates for correlation parameters $\hat{\rho}_{xij}$ and $\hat{\rho}_{yij}$. It allows us to use a filter not only for noise removing but also as an element of algorithms for texture-correlation analysis, for example, when segmenting images.

The proposed filtering algorithm is close in structure to the Kalman vector filter [30]. Figure 6.9 shows the dependences of the matrix element in the case of using the Kalman vector filter and nonlinear doubly stochastic filtering.

It should be noted that the nonlinear filter slightly loses the Kalman filter at the beginning of the line during the stabilization process. As the number of elements increases, convergence to the Kalman filter is observed. The result obtained allows us to state that the procedures are close to each other when filtering homogeneous images with known correlation parameters. The situation changes significantly when filtering heterogeneous images. Figure 6.10 shows the results of the satellite image filtering.

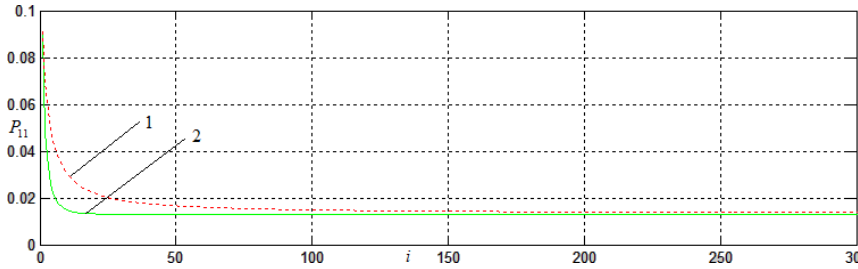


Figure 6.9. Error dispersions for a nonlinear filter (1) and linear Kalman filtering (2).

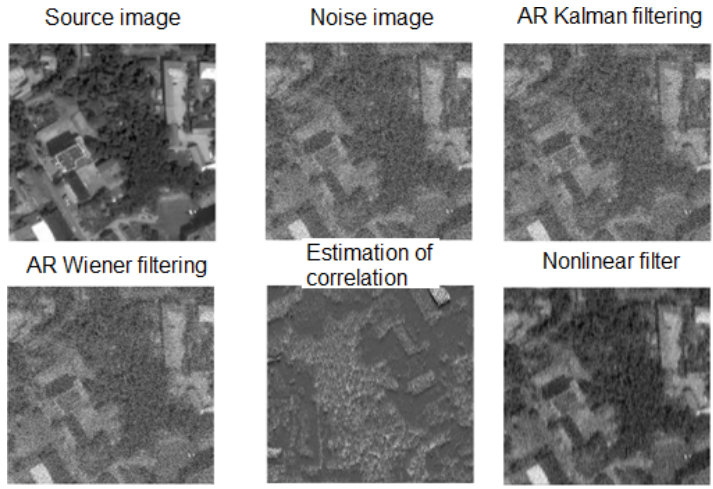


Figure 6.10. Satellite image filtering.

For the Kalman filter, the average dispersion of the filtering errors for the signal-to-noise ratio = 2 was 0.4981, for the discrete Wiener filter, it was 0.4978, and for the proposed algorithm error dispersion was 0.1359. Figure 6.11 shows the filtering process performed on a separate line of the image. The graphs show the source image line (1), the row of the noisy image (2), the result of the Kalman filter processing (3), the result of Wiener filtering (4), and the proposed nonlinear filter applying (5).

The adaptive behavior of the proposed nonlinear filtering algorithm is clearly visible on the graphs.

In this section, algorithms for nonlinear filtration of two-dimensional images are synthesized and investigated. The algorithms

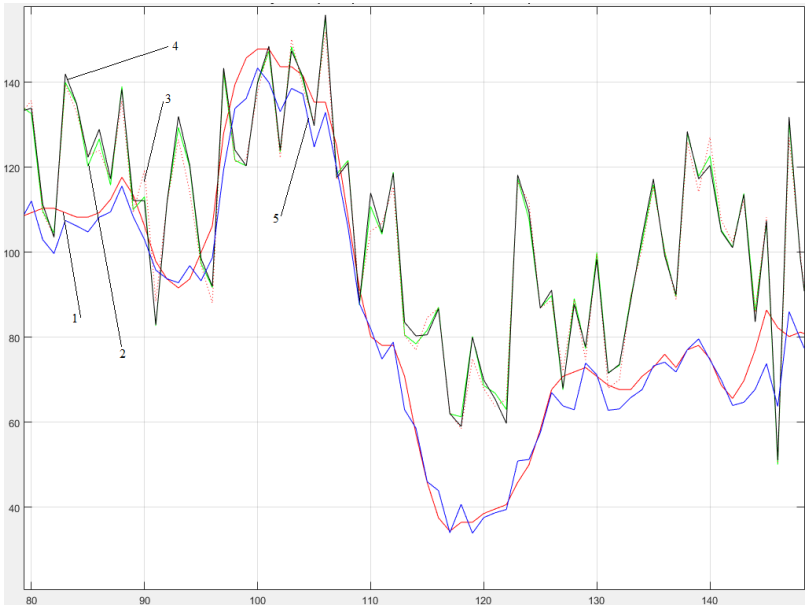


Figure 6.11. Filtering a separate line of the image.

provide estimation of not only the basic RF but also the field of the correlation parameters. The analysis of the obtained algorithms showed that for heterogeneous images, developed algorithms exceed the known based on the assumption of image homogeneity.

6.4 Detection of Anomalies on the Background of Noise

In many applications, it is often desirable to detect anomalies that may appear on a recurrent multidimensional image of frame sequence. For example, these anomalies may be forest fires, pathological changes in medical images, new objects in a security area, etc.

6.4.1 *Optimal algorithms for signal detection*

Let observations of RF $x_{\bar{j}}^{t}$ with space-time correlation relationship and additive noise RF $\theta_{\bar{j}}^{t}$, which consists of independent random

variables with zero means and variances V_θ, be made

$$z_{\bar{j}}^t = x_{\bar{j}}^t(\bar{\chi}_t) + \theta_{\bar{j}}^t, \quad \bar{j} \in J_t, \quad t = 1 \dots k. \tag{6.38}$$

Parameter vector $\bar{\chi}_t$ lets us take into account, for example, possible mutual spatial displacements and rotations of nearest-neighbor image frames.

Let the appearance of a deterministic signal cause changes in the model (6.35) only in index domain $\bar{j} \in G \subset J$ (Figure 6.12). In particular, domain G can be a part of the last frame observed:

$$z_{\bar{j}}^k = s_{\bar{j}} + x_{\bar{j}}^k(\bar{x}_t) + \theta_{\bar{j}}^k, \quad \bar{j} \in G, \tag{6.39}$$

where $\{s_{\bar{j}}, \bar{j} \in G\}$ are the values of a useful (detectable) signal.

Under considered circumstances, it is necessary to find out a rule to test the hypothesis H_0 on the absence of anomalies in the domain G under alternative assumption H_1 on the validity of model (6.39).

At predetermined probabilistic characteristics of model components (6.38) and (6.39), it is possible to estimate the corresponding conditional PDF of observations $W(Z|H_0)$ and $W(Z|H_1)$. Therefore, to solve the detection problem, one should compare the results with the threshold level Λ_0 of likelihood ratio (LR):

$$\Lambda = \frac{W(Z|H_1)}{W(Z|H_0)} \begin{cases} \geq \Lambda_0 - \text{signal}, \\ < \Lambda_0 - \text{no signal}. \end{cases} \tag{6.40}$$

To simplify the calculations, let's represent conditional PDF in a product form: $W(Z|H_{0,1}) = W(Z_0|H_{0,1})W(Z_G|Z_0, H_{0,1})$, where Z_G is a set of observations in domain G and Z_0 is a set of all observations, which do not belong to the domain of the intended signal. Since

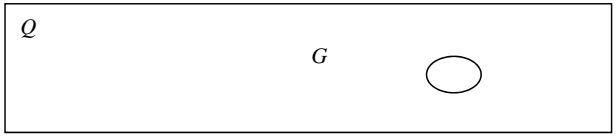


Figure 6.12. Domain G, where and only where a signal becomes apparent, Q — a complimentary domain.

$W(Z_0|H_0) = W(Z_0|H_1)$, LR in (6.40) can be rewritten in the form

$$\Lambda = \frac{W(Z_G|Z_0, H_1)}{W(Z_G|Z_0, H_0)}. \tag{6.41}$$

Let's approximate conditional PDF, included in LR (6.38), using Gaussian distribution:

$$W(Z_G|Z_0, H_{0,1}) = \frac{1}{(2\pi)^{N/2}\sqrt{\det V_{0,1}}} \exp\left(-0.5||z^k - m_{0,1}||^2_{v_{0,1}^{-1}}\right), \tag{6.42}$$

where $m_0 = \{m_{0\bar{j}}\}$, $V_0 = \{V_{0\bar{l}\bar{j}}\}$ and $m_1 = \{m_{1\bar{j}}\}$, $V_1 = \{V_{1\bar{l}\bar{j}}\}, \bar{i}, \bar{j} \in G$, are conditional expectations and spatial covariance matrices of observations Z_G in the absence and presence of a useful signal, correspondingly.

Using equations (6.37) and (6.38), we obtain the following formulae for the conditional mean values: $m_{0\bar{j}} = \widehat{x}^k_{э\bar{j}}, m_{1\bar{j}} = s_{\bar{j}} + x^k_{э\bar{j}}, \bar{j} \in G$, where $\widehat{x}^k_{э\bar{j}} = M\{x^k_{\bar{j}}|Z_0\}$ is the optimal prediction of RF values $x^k_{\bar{j}}, \bar{j} \in G$, obtained on the basis of all available observations Z_0, which do not belong to a signal domain. Matrices V_0 and V_1 are the same: $V_0 = V_1 = V = P_э + V_\theta$, where $P_э$ is a covariance matrix of optimal prediction errors. Substituting the obtained formulae in (6.41) and (6.42) and taking logarithms, it is possible to find the following algorithm for signal detection:

$$L = \sum_{\bar{l},\bar{j}\in G} s_{\bar{j}} V^{-1}_{\bar{l}\bar{j}} \left(z^t_{\bar{j}} - \widehat{x}^t_{э\bar{j}}\right) \begin{cases} \geq \Lambda_0 - \text{signal}, \\ < \Lambda_0 - \text{no signal}, \end{cases} \tag{6.43}$$

where $\lambda = \ln\Lambda_0 + 0,5\|s_{\bar{j}}\|^2_{V^{-1}}$ is a detection threshold. Thus, the procedure of anomaly detection includes compensation of interfering images by subtracting optimal prediction $\widehat{x}^k_{э\bar{j}}$ from observation $z^k_{\bar{j}}$. The prediction is found on the basis of all observations and does not belong to domain G. Let's call this kind of prediction "domain prediction" (Figure 6.13(a)). After compensation of interference RF linear weighted summation of all residuals is carried out $D = z^k_{\bar{j}} - \widehat{x}^k_{э\bar{i}}$.

Another form of optimal detection procedure can be obtained assuming that any useful signal can occupy all image frames, i.e.

domain G includes all multidimensional grids $J_1, J_2, \ldots, J_t$. Then, the best prediction $x^k_{\bar{\jmath}} \equiv 0$, $\bar{j} \in G$, and algorithm (6.41) can be written as follows:

$$L = \sum_{\bar{l},\bar{j} \in G} s_{\bar{l}} V^{-1}_{\bar{l}\bar{j}} z^k_{\bar{j}} \begin{cases} \geq \Lambda_0 - \text{signal}, \\ < \Lambda_0 - \text{no signal}, \end{cases} \tag{6.44}$$

where $V_{\bar{l}\bar{j}} = V_{x\bar{l}\bar{j}} + V_\theta E_{\bar{l}\bar{j}}$, $V_{x\bar{l}\bar{j}}, \bar{l}, \bar{j} \in G$ is a covariance matrix of an interfering image.

Direct implementation of procedure (6.44) is difficult because of the large number of computations. However, expanding the spatial symmetric matrix into the product of two triangle ones $V^{-1}_{\bar{l}\bar{j}} = A_{\bar{l}\bar{\nu}} A_{\bar{\nu}\bar{l}}, \bar{l}, \bar{\nu}, \bar{j} \in G$, it is possible to present (6.42) in the form

$$L = \sum_{\bar{l},\bar{j},\bar{\nu} \in G} s_{\bar{l}} A_{\bar{l}\bar{\nu}} A_{\bar{\nu}\bar{j}} z^t_{\bar{j}} \begin{cases} \geq \Lambda_0 - \text{signal}, \\ < \Lambda_0 - \text{no signal}, \end{cases} \tag{6.45}$$

which corresponds to the preliminary decorrelation of image sequences $z^t_{\bar{j}}, \bar{j} \in G$, and subsequent weighted summation with weights $s_{\bar{l}} A_{\bar{l}\bar{\nu}}$. In many cases, such an approach helps find acceptable practice quasi-optimal algorithms of decorrelation. At the same time, an independent analysis of adaptive recursive filters, whose properties are similar to decorrelation ones, is an important area to look for relatively simple technical or software implementation of system for image sequence processing. New quality of algorithms (6.42) and (6.43) in comparison to (6.41) is the division of time-consuming operation of optimal prediction or decorrelation, which is not connected with a signal form, and relatively simple weighted summation, which takes into account the kind of a useful signal. It allows us to solve both problems of anomaly detection with unknown location and

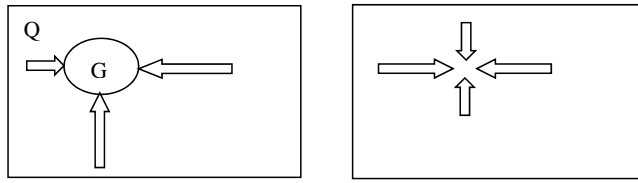


Figure 6.13. (a) domain prediction; (b) point prediction.

more complex problems of multialternative detection (recognition) of several signal types rather easily.

There is one more form of decision rule, which gives the same result [30]:

$$L = \sum_{\bar{j}\in G} s_{\bar{l}}\delta_{\bar{j}} = \sum_{\bar{j}\in G} s_{\bar{l}}\left(z_{\bar{j}}^{k} - \tilde{z}_{\bar{j}}^{k}\right)/\sigma_{\bar{j}}^{2} \begin{cases} \geq \Lambda_0 - \text{signal}, \\ < \Lambda_0 - \text{no signal}, \end{cases} \tag{6.46}$$

based on the ⟨point prediction⟩. At the same time, observation prediction $\tilde{z}_{\bar{j}}^{k} = M[z_{\bar{j}}^{k}|(Z\backslash z_{\bar{j}}^{k})]$ is made taking into account all other predictions from Z and assuming that there is no signal (Figure 6.13(b)).

Despite the equality of statistics (6.44) and (6.46), there is a fundamental difference between them. In (6.46), prediction and compensation are made according to observations that do not have a signal, so if there is a signal in G, it will be distorted only by prediction errors. If these errors are small, the compensation residuals D will be close to the values of detectable S_G (it is possible to see the signal with little distortion). In (6.46), while constructing a point prediction, all other observations are used, including those with a signal. Therefore, in the residuals Δ_G of this compensation, every signal value will be distorted not only by prediction errors of an interfering image but also by other signal elements as well. Even if prediction errors are small, a much distorted signal will be observed.

A significant drawback of all considered algorithms is quite a complex analysis of signal detection efficiency. While searching the ways to simplify this analysis, we managed to obtain one more procedure to detect stretched signals [43]. After elementary transformations and considering that $P(E+V_{\theta}^{-1}P_{\ni}) = P_{\ni}$, $z_l - \hat{x}_{\ni j} = (E+V_{\theta}^{-1}P_{\ni})(z-\hat{x})$, we can find the following decision rule form [14]:

$$L = s_l^t V_{\theta lj}^{-1}\left(z_j^t - \hat{x}_j^t\right) \begin{cases} \geq \Lambda_0 - \text{signal}, \\ < \Lambda_0 - \text{no signal}, \end{cases} \tag{6.47}$$

which contain the optimal estimates $\hat{x}_j^t$.

Thus, at Gaussian distribution approximation, the obtained detection procedure suggests optimal RF filtration, calculation of covariance matrix of filtering errors, and weighted summation using (6.45). Since algorithm (6.46), in contrast to known detectors, does not require time-consuming calculation of covariance matrices of prediction errors, it can be used not only in image processing system but also for probability analysis of detection efficiency.

These results allow us to specify the conditions under which a proposed replacement of conditional PDF by normal distribution is valid. First of all, it is a wide class with Gaussian models (6.38) and (6.39). In these cases, procedures (6.43)–(6.45), (6.46), and (6.47) are optimal. When models (6.37) and (6.38) components are non-Gaussian, the sufficient optimum condition is the possibility to approximate the posterior PDF prediction $x^k_{\ni\bar{j}}$ using normal distribution. Let's note that the last condition is fulfilled in many applied tasks of RF processing with significant space-time correlative relationship and is usually equivalent to the high posterior prediction accuracy.

6.4.2 *Efficiency of anomaly detection*

In the Gaussian approximation, conditional statistics distributions (6.42)–(6.44) will also be Gaussian. Therefore, to calculate the efficiency of anomaly detection, it is sufficient to find conditional moments of statistics (6.42):

$$M\{L/H_0\} = 0, \quad M\{L/H_1\} = S_{\bar{e}} V^{-1}_{\bar{j}\bar{e}} S_{\bar{j}}, \quad \sigma^2_L = S_{\bar{e}} V^{-1}_{\bar{j}\bar{e}} S_{\bar{j}}.$$

Thus, at a given quantile x_F of level P_F of a normal distribution, a detection threshold $\lambda = x_F \sigma_L$ and probability of correct detection can be determined:

$$P_D = 0,5 + \Phi_\circ(\Delta_D - x_F), \tag{6.48}$$

where

$$\Delta_D = \sqrt{S_{\bar{e}} V^{-1}_{\bar{j}\bar{e}} S_{\bar{j}}}, \bar{j}, \bar{e} \in G; \quad \Phi_\circ(x) = (1/\sqrt{2\pi}) \int_0^x \exp(-z^2/2) dz,$$

$$P_{\ni\bar{j}\bar{e}} = (E_{\bar{j}\bar{q}} - P_{\bar{j}\bar{\nu}} V^{-1}_{\theta\bar{\nu}\bar{q}})^{-1} P_{\bar{q}\bar{e}}.$$

It can be shown that there is a simple relationship between the spatial covariance matrices $P_{\ni\bar{j}\bar{e}}$ of optimal prediction errors and $P_{\bar{j}\bar{e}} = M\{(x^t_{\bar{j}} - \widehat{x}^t_{\bar{j}})(x^t_{\bar{e}} - \widehat{x}^t_{\bar{e}})\}$ of optimal linear estimation errors $\hat{x}^t_{\bar{j}} = \hat{x}^t_{\bar{j}}(Z)$. Indeed, analyzing the Wiener–Hopf equation for multidimensional discrete RF, it is possible to obtain the following equation:

$$P_{\ni\bar{j}\bar{e}} = (E_{\bar{j}\bar{q}} - P_{\bar{j}\bar{\nu}} V^{-1}_{\theta\bar{\nu}\bar{q}})^{-1} P_{\bar{q}\bar{e}},$$

where $V_{\theta\bar{\nu}\bar{q}} = \sigma^2_\theta E_{\bar{\nu}\bar{q}}$. Substituting the obtained link between matrices in (6.47), we can find a formula for efficacy calculation of signal

detection:

$$\Delta_D = \sqrt{S_{\bar{e}} V_\theta^{-1} (E_{\bar{e}\bar{j}} - V_\theta^{-1} P_{\bar{e}\bar{j}}) S_{\bar{j}}}. \tag{6.49}$$

If there are no interfering images $x_{\bar{j}}$ or if they are estimated precisely $P_{\bar{e}\bar{j}} = 0$, the deterministic signal detection performance on the background of white noise is determined by a well-known signal/noise ratio $\Delta_D = \sqrt{\sum_{\bar{j}\in G} S_{\bar{j}}^2 / V_\theta}$. Errors $x_{\bar{j}}^t$ with time-space correlation decrease the detection accuracy, which is measured by the non-zero covariance matrix of optimal RF estimation $x_{\bar{j}}^t$ on the background of white noise $\theta_{\bar{j}}^t, \bar{j} \in J$.

Let's consider two important examples of how it is possible to calculate the potential efficiency of point anomaly detection.

Suppose, that a point signal $s_{\bar{j}} = s_0$ is being detected. It occupies one grid G element J_t. In this case,

$$\Delta_D = \sqrt{Q(1 - \sigma_\varepsilon^2 / \sigma_\theta^2)},$$

where $Q = s_0^2/\sigma_\theta^2$ is a signal/noise ratio and σ_ε^2 is the error variance at optimal estimation of one RF element $x_{\bar{j}}^t$, $\bar{j} \in G$, on the basis of all available observations $z_{\bar{j}}^t$, $\bar{j} \in J_t$, $t \in T$. Magnitude of error variance σ_ε^2 can be obtained by known methods of RF optimal estimation theory, which are observed on the background of noise [18, 21].

Let's now consider the task of finding a point object based on the results of observations of imagery $M_3 M_1 \times M_2$ size. Such a problem can arise, for example, during multizone observations of one and the same object using different systems functioning in different spectral range [4–8, 19]. In this case observation model (6.38) can be written as following

$$z_{\bar{j}} = x_{\bar{j}} + \theta_{\bar{j}}, \quad \bar{j} = (j_1, j_2, j_3); \quad j_k = 1 \ldots M_k.$$

Let's assume that datasets $z_{\bar{j}}$, $j_3 = 1 \ldots M_3$ are spatially shifted; therefore, the appearance of the desired signal increases the RF level by s_0 in one and the same element in each image frame of which is numbered (j_1^0, j_2^0) (Figure 6.15). Covariance functions of interfering images can be written as follows:

$$\begin{aligned} R_x(m_1, m_2, m_3) &= M\left\{x_{j_1,j_2,j_3} x_{(j_1+m_1),(j_3+m_3),(j_3+m_3)}\right\} \\ &= R_1(m_1, m_2)\rho^{1(m_3)}, \end{aligned}$$

where $R_1(m_1, m_2) = R_x(m_1, m_2, m_3 = 0)$ is one and the same covariance function from M_3 image frames, $1(m_3) = 1$, if $m_3 \neq 0$,

and $1(m_3 = 0) = 0$. Note that such representation means equality of correlation distances between any pair from M_3 analyzed image frames. After simple calculations, it's possible to say that formula (6.49) for a given task can be transformed into

$$\Delta_D = \sqrt{M_3} Q \sqrt{1 - F_R(\bar{0}) \left\{ \frac{(1 + (M_3 - 1)\rho) G_1(\omega_1, \omega_2)}{\sigma_\theta^2 + (1 + (M_3 - 1)\rho) G_1(\omega_1, \omega_2)} \right\}},$$

where $Q = s_0/\sigma_\theta^2$; $G_1(\omega_1, \omega_2) = F\{R_1(\omega_1, \omega_2)\}$; $F(\omega_1, \omega_2)\{\cdot\}$ and $F_R(m_1, m_2)\{\cdot\}$ are direct and inverse two-dimensional discrete Fourier transformations and $F_R(\bar{0})\{\cdot\} = F_R(m_1 = 0, m_2 = 0)\{\cdot\}$.

The analysis of the obtained equation leads to the following important conclusions about the effectiveness of the point anomaly optimal detection on an arbitrary number of simultaneously processed images.

In the absence of interframe correlation ($\rho = 0$), co-processing M_3 of frames yields to gain a threshold signal in $\sqrt{M_3}$ times in comparison to signal detection on one image frame notwithstanding the covariance function $R_1(m_1, m_2)$ and size $M_1 \times M_2$ of a frame.

Interframe correlation ($\rho \neq 0$) leads to losses in detection efficiency, which correspond to additional increase in variance σ_x^2 of interfering images in $(1 + (M_3 - 1)\rho)$ times.

The above-mentioned properties make it easy to recalculate the characteristics of point anomaly detection on a single image frame in the case of co-processing of arbitrary number of mutually correlated image frames.

6.4.3 *Extended signals' detection against a doubly stochastic background*

Equation (6.47) shows that the best detection procedure involves the optimal filtration of the RF, the calculation of the covariance matrix of the filtering errors, and the execution of the weighted summation in accordance with the indicated formulas. The most complex of these steps is the filtration of the RF. This is due to the fact that real satellite imagery has a pronounced spatial heterogeneity. Using standard optimal linear filters for such images leads to significant errors. The solution to this problem is possible due to the use of special filters that take into account the complex nature of the images. Consider the synthesis of such filters for the case where correlation

between individual components of a polyzonal image can be ignored. In this case, the processing of the image component can be carried out independently of one another. The conducted studies [41, 43, 46] show that to form such filters, it is possible to use doubly stochastic image models, which allow describing inhomogeneous signals [46]. As an example, consider model (6.36). Note that this model imitates inhomogeneous images [41], which allows us to recommend it for describing real satellite images. In this case, we can use the vector (row by row) nonlinear Kalman filter (6.37) to reduce the noise [41]. The use of this algorithm is possible under the condition of precisely known characteristics of the information RF. So we need to know coefficients r_{1x}, r_{2x}, r_{1y}, r_{2y} and also parameters ρ_{0x}, ρ_{0y} and $\sigma^2_{\rho x}$, $\sigma^2_{\rho y}$, σ^2_x. Otherwise, a preliminary evaluation of these parameters is necessary. For this, pseudo-gradient procedures can be used, as well as expressions for CF of doubly stochastic RF models [41].

Let's compare two detectors of anomalies constructed on the basis of a doubly stochastic model (Algorithm 1) and on the basis of the usual autoregressive model [4] (Algorithm 2). Further, by Algorithm 1, we will understand an algorithm involving the evaluation of images based on a nonlinear doubly stochastic model. And by Algorithm 2, we will understand an algorithm based on image estimation based on a linear autoregressive model. We consider this designation reasonable and understandable in the context of the chapter. In this case, the detection will be performed on real images obtained from the LandSat-8 satellite. Studies are conducted for three images. We choose four areas for each image, where an anomaly may occur. It should be noted that the areas are selected based on the structure of the images to be examined, taking into account the greater and smaller heterogeneity, and the detection procedures are performed not for the entire image but only for these areas. Figures 6.14(a)–6.14(c) show examples of images with signals located in different parts of the images and also reflect the probabilities of correct detection obtained using two algorithms. The sizes of all images are 250×250. The images are distorted by white Gaussian noise with a single dispersion. The size of the square is 4×4, and the radius of the circle is 2. The signal-to-noise ratio is 1. The statistics are removed 150 times.

Table 6.1 shows the gain of Algorithm 1 in relation to Algorithm 2 for the magnitude of the threshold signal when the probability of correct detection is 0.5 and the probability of false alarm is 0.001. It corresponds to the threshold $L_0 = 3,1\sigma^2_{z1}$.

Table 6.1. Gain (in percent) of the proposed detection algorithm based on a doubly stochastic model in comparison with the detection algorithm based on the AR model.

Shape/Image	**Location 1**	**Location 2**	**Location 3**	**Location 4**
Square in Image 1	0	0	0	0
Circle on Image 1	5	2	0	2
Square in Image 2	68	3	13	4
Circle on Image 2	60	4	3	5
Square in Image 3	21	4	4	5
Circle on Image 3	70	5	7	7

Analysis of the results shows that the algorithm based on the doubly stochastic model works better than the algorithm based on the autoregressive model and provides reliable detection of the signal in 90–95% of the cases. The small values of the gains in Table 6.1 are explained by the fact that the signals have small dimensions, and their neighborhoods are on a comparable scale to homogeneous ones. If the signal is "at the junction" of homogeneous regions, an algorithm based on a doubly stochastic model provides a significant (up to 70%) gain in the signal level term. The gains presented in Table 6.1 are calculated for each case from the expression

$$\text{Gain} = \frac{Pd_{\text{ds}} - Pd_{\text{ar}}}{Pd_{\text{ar}}},$$

where Pd_{ds} and Pd_{ar} are percentages of correctly detected signals based on doubly stochastic and autoregressive models, respectively.

Analyzing Figure 6.14, we can conclude that we also have gains in correct detection probability terms on equal signal-to-noise ratios. Furthermore, the probability of correct detection depends not only on the shape and sizes of the signal itself but also on the brightness values in its immediate neighborhood. In this sense, a more universal algorithm is an algorithm based on doubly stochastic RF models.

6.5 Image Alignment

While synthesizing algorithms for solving different tasks of frame sequence processing $z^1, z^2, \ldots$, it is usually assumed that observations $z^1_{\bar{j}}, z^2_{\bar{j}}, \ldots$ in the nod $\bar{j}$ of a grid Ω correspond to the

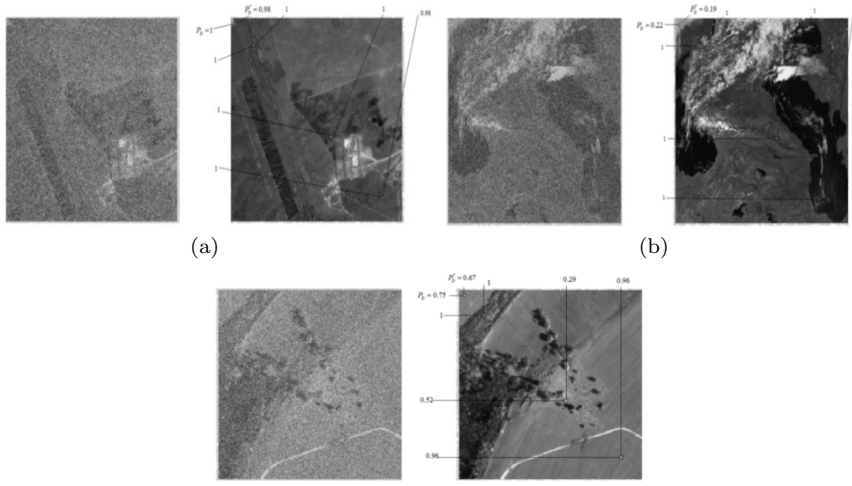


Figure 6.14. The noisy image (left) and the source images (right) with the probabilities of correct detection of a square signal: on the left, the probabilities for Algorithm 1 are presented and on the top, the probabilities for Algorithm 2 are presented.

same space point. In the real world, receiver motion, its imperfection, and other factors lead to the fact that $z_j^1, z_j^2, \ldots$ correspond to the different space points. As a result, the sequence $z^1, z^2, \ldots$ will demonstrate interframe displacement from frame to frame, i.e. interframe geometric transformation (IGT), such as shifts and rotations. If such distortions are not taken into account, the efficiency of processing algorithms can dramatically decrease.

6.5.1 *Tensor shift filtering*

Tensor Kalman filter described in Section 2.3, which estimates frame sequence defined by tensor model, can be extended to joint estimation of frames and interframe shift of these frames. Let's consider this extension.

Let a sequence of m-dimensional frames be defined by a linear tensor stochastic equation

$$x^t = \rho^t x^{t-1} + \vartheta^t \xi^t, \quad t = 1, 2, \ldots.$$

Their observations look like

$$z^t = x^t(y^t) + \theta^t, \tag{6.50}$$

where $y^t = (y_1^t, \ldots, y_m^t)$ is an IGT parameter vector of the tth frame, $x^t(y^t)$ is a proper observation of a frame x^t with parameters y^t, and θ^t is a white Gaussian RF of observation noise. Thus, the sequence of shift vectors is also described by a linear stochastic equation

$$y^t = \Im^t y^{t-1} + \Phi^t \eta^t, \quad t = 1, 2, \ldots,$$

where $\Im^t, \Phi^t$ are $(m \times n)$-matrices (second-rank tensor) and $\eta^t = (\eta_1^t, \ldots, \eta_m^t)$ is the white Gaussian generating vector of a shift model.

Using observations (6.50), it is necessary to find an estimation of the successive RF frame x^t and estimation y^t when obtaining a recurrent observation z^t. To find this estimation, let's use tensor filtering equations as in (6.35). In these equations, both the frame itself x^t and its parameters y^t are included in the estimated frame x^t. This integration is a component tensor, so filtering equations become a bit more complicated. Omitting the intermediate calculations, we give the resulting algorithm of recurrent RF estimation and displacement [43]:

$$\begin{aligned}
\widehat{x}^t &= \widehat{x}_э^t + P_x^t (V_\theta^t)^{-1} \left(z^t - \widehat{x}_э^t\right) + P_B^t (V_\theta^t)^{-1} \frac{d\widehat{x}_э^t}{d\alpha^t} \left(z^t - \widehat{x}_э^t\right), \\
\widehat{y}^t &= \widehat{y}_э^t + P_y^t (V_\theta^t)^{-1} \frac{d\widehat{x}_э^t}{d\alpha^t} \left(z^t - \widehat{x}_э^t\right) + (P_B^t)^T (V_\theta^t)^{-1} \left(z^t - \widehat{x}_э^t\right),
\end{aligned} \tag{6.50}$$

where $\widehat{x}_э^t = x^t(\widehat{y}_э^t)$, $\widehat{y}_э^t = \Im^t \widehat{y}^{t-1}$.

Besides computational problems, it is connected with model specification (6.50), namely, defining function $x^t(y^t)$, i.e. how the frame x^t looks like at IGT parameters y^t. However, if IGT kind is given (for example, shift or rotation), it is much easier to define the function $x^t(y^t)$.

Let's examine the task of RF alignment when parameter distribution for IGT is not given. For this purpose, let's consider the IGT model first.

Let RF $\dot{X} = \{x_{\bar{u}}^i : \bar{u} \in U,\ i = 1, 2, \ldots\}$ be set on some continuous domain U at any specific time i. Each frame $x^i = \{x_{\bar{j}}^i : \bar{j} \in \Omega_i\}$ of a

grid field X is a system of frame $x^i = \{x^i_{\bar{u}} : \bar{u} \in U\}$ values on the grid $\Omega_i = \{\bar{j} : (j_1, \ldots, j_n) : j_k = \overline{1, M_k}\}$. Besides, the position and shape of grids Ω_i can vary with time, while index size $M_1 \times M_2 \times \cdots \times M_n$ remains constant. Figure 6.18 demonstrates only some possible positions of two-dimensional grids Ω_{i-1} (continuous line) and Ω_i (dashed line): Figure 6.15(a) — grid Ω_i is obtained from a rectangular grid Ω_{i-1} by a parallel shift; Figure 6.15(b) — grid is obtained by shift and rotation; Figure 6.15(c) — grid Ω_i is curvilinear; Figure 6.15(d) — both grids are curvilinear.

Each of the grids Ω_i may be considered as a system of coordinates, so the task of RF alignment can be restated as a task of finding of transformation of coordinates of grid Ω_i nods in the system of coordinates Ω_u of domain U. Sometimes (e.g. to compensate for the noise), it is easier to find coordinate Ω_i transformation into Ω_{i-1}, i.e. to align each recurrent frame x^i with proceeding frame x^{i-1}.

Let's consider the task of aligning two frames x^{i-1} and x^i. In a general case (Figure 6.15(d)), it's necessary to estimate grid form Ω_{i-1} and find the transformation of coordinates Ω_i into Ω_{i-1}.

Let an observation model of field X have the following form:

$$z^i = x^i + \theta^i, \quad i = 1, 2, \ldots, \tag{6.51}$$

where $\theta = \{\theta^i_{\bar{j}}\}$ is a field of independent RV. Then, if RF $\dot{X}$ is stationary, a conditional mutual PDF $w(z^{i-1}, z^i | f)$ can be obtained, where f is a transformation of coordinates Ω_i in (rectangular) coordinate system Ω_{i-1}. This makes it possible to apply various statistical estimations to combine a pair of frames x^{i-1} and x^i, for example, maximum likelihood estimate

$$f = \arg\max_f w(z^{i-1}, z^i | f). \tag{6.52}$$

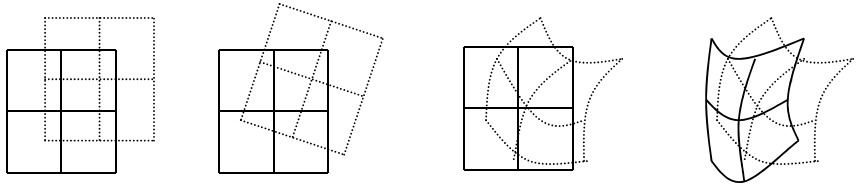


Figure 6.15. Positions of grids of two sequenced frames: (a) parallel shift; (b) shift and rotation; (c) one grid is curvilinear; (d) both grids are curvilinear.

In the general case of transformations f all coordinates $f(\bar{j})$ of all frame values x^i should be estimated in the system Ω_{i-1}. Therefore, function f in (6.52) contains a large number of parameters and calculation becomes rather complicated.

Alignment task is sufficiently simplified if one has to take into account the type of transformation f. Then, it is desirable only to determine its parameters $\bar{\alpha}$. In such cases, estimation (6.52) takes the following form:

$$\hat{\bar{\alpha}} = \arg\max_{\bar{\alpha}} w(z^{i-1}, z^i|\bar{\alpha}) \tag{6.53}$$

and contains only a few parameters. Estimation (6.53) can be used in the case of RF *a priori* parametric uncertainty $\dot{X}$ and observation model (6.51). For this, parameter vector $\bar{\alpha}$ must be supplemented by RF unknown parameters $\dot{X}$ and an observation model.

6.5.2 *Alignment of two frames of Gaussian random field*

Let the Gaussian stationary field $\dot{X}$ have a zero mean and CF

$$V(i, \bar{u}) = M[\dot{x}^l_{\bar{v}} \dot{x}^{l+i}_{\bar{v}+\bar{u}}] = \sigma_x^2 \rho(|i|, |u_1|, \dots |u_n|), \tag{6.54}$$

Where $\rho(i, |u_1|, \dots |u_n|)$ is the field $\dot{X}$ correlation coefficient at a distance $|i|$ along the time axis and at a distance $|u_k|$ along the kth spatial axis. In observation model (6.51), noises θ will also be assumed as Gaussian with zero mean and constant value σ_θ^2.

Let there exist some observations z^{i-1} of a frame x^{i-1} at the nodes of a rectangular grid Ω_{i-1} with a unit step and some observations z^i of a frame x^i at the grid Ω_i nodes. It is necessary to estimate parameters $\bar{\alpha}$ of IGT x^{i-1} and x^i, i.e. to find the estimation $\hat{\bar{\alpha}}$ of parameters in $f(\bar{j}, \bar{\alpha})$.

If we assume that $i = 1$ and choose the grid Ω_1 axis coinciding with the coordinate axis in which CF (6.54) is set, then the Gaussian joint PDF of a frame x^1 and its observations can easily be calculated from (6.51) and (6.54). If parameter vector $\bar{\alpha}$ is given, grid Ω_2 position relative to Ω_1 becomes definite, therefore, it is possible to find joint conditional PDF of observations $Z = (z^1, z^2)$ at

given $\bar{\alpha}$:

$$w(Z|\bar{\alpha}) = \frac{1}{(2\pi)^{k/2}\det^{1/2}(V(\bar{\alpha}))}\exp\left(-\frac{1}{2}ZV^{-1}(\bar{\alpha})Z\right), \qquad (6.55)$$

where $V(\bar{\alpha})$ is a covariance matrix of observations Z; k is a number of elements in Z. It should be noted that calculation of (6.55) maximum is rather a complicated task which is absolutely unrealizable in real-time systems.

In order to simplify the task, let's consider the estimation, which can be obtained as a result of only exponent maximization in (6.55):

$$\hat{\bar{\alpha}} = \arg\min_{\bar{\alpha}} ZV^{-1}(\bar{\alpha})Z = \arg\min_{\bar{\alpha}} J(Z,\bar{\alpha}). \qquad (6.56)$$

Functional $J(Z,\bar{\alpha})$ is the Mahalanobis distance of sample Z from the origin for covariance matrix $V(\bar{\alpha})$. Thus, estimation (6.56) minimizes the Mahalanobis distance of observed Z from the origin. Let's consider this distance in detail. From (6.43), it follows that the following observation takes place:

$$J(Z,\bar{\alpha}) = ZC^{-1}(\bar{\alpha})(Z - Z^*(\bar{\alpha})) = ZC^{-1}(\bar{\alpha})\Delta_Z^*(\bar{\alpha}), \qquad (6.57)$$

where $Z^*(\bar{\alpha})$ is an optimal (in our case linear) prediction of observations Z into a point, $\Delta_Z^*(\bar{\alpha})$ are prediction errors, i.e. compensation residues into a point, and $C(\bar{\alpha})$ is a diagonal matrix of error variance $\Delta_Z^*(\bar{\alpha})$. If observations z^1 and z^2 are used twice ($z^1 \to z^2$ and $z^2 \to z^1$), the following formula can be obtained

$$J(z^1,z^2,\bar{\alpha}) = [z^2 - \hat{z}^2(\bar{\alpha})]N_2^{-1}(\bar{\alpha})[z^2 - \hat{z}^2(\bar{\alpha})], \qquad (6.58)$$

what is Mahalanobis distance between observations z^2 and their predictions $\hat{z}^2(\bar{\alpha})$ according to z^1, where $N_2(\bar{\alpha})$ is the error covariance $\hat{\Delta}_2(\bar{\alpha})$ of this prediction.

Experimental studies show poor estimation quality obtained using Mahalanobis distance. It can be explained by the fact that in (6.57)–(6.58) compensation residues and matrices inversed to $C(\bar{\alpha})$ or to $N_i(\bar{\alpha})$ are multiplied. In addition, compensation residues can decrease along $\bar{\alpha}$ only to a certain limit, so minimization of these expressions can occur mainly due to an increase in matrix elements $C(\bar{\alpha})$ or $N_i(\bar{\alpha})$. This does not happen while using MLM estimates, as

PDF (6.55) is inversely proportional to the square root of covariance matrix determinant $V(\bar{\alpha})$.

Much better IGT estimates can be obtained by minimizing the compensation residues $\Delta_Z^*(\bar{\alpha})$, $\hat{\Delta}_1(\bar{\alpha})$, or $\hat{\Delta}_2(\bar{\alpha})$. It is easier to minimize residues

$$\hat{\Delta}_2(\bar{\alpha}) = z^2 - \hat{z}^2(\bar{\alpha}),$$

as the optimal compensation of the recurrent frame z^2 according to the previous frame z^1 is determined. It can often be the main alignment task. Using this approach, it is possible to obtain the following estimates

$$\hat{\bar{a}} = \arg\min_{\bar{\alpha}} M\left[\sum_{\bar{j}} |z_{\bar{j}}^2 - \hat{z}_{\bar{j}}^2(\bar{\alpha})|\right], \quad \hat{\bar{a}} = \arg\min_{\bar{\alpha}} M\left[\sum_{\bar{j}} (z_{\bar{j}}^2 - \hat{z}_{\bar{j}}^2(\bar{\alpha}))^2\right], \tag{6.59}$$

and a number of other estimates, which depend on the used metric. Further, a variety of estimates can be obtained using various predictions $\hat{z}_{\bar{j}}^2(\bar{\alpha})$, for example, optimal prediction or various interpolations of observation z^1, initially set only on grid Ω_1.

Generally speaking, it should be mentioned that compensation estimates (6.59) are biased. They do not estimate IGT parameters $\bar{\alpha}$ but only optimize selected compensation in terms of a certain metric. Thus, frame z^1 and z^2 alignment with the use of estimated parameters $\bar{\alpha}$ is pseudo-alignment in the sense of the best compensation of the selected type. However, in case of a good choice of prediction function, rather effective alignment is often provided.

6.5.3 *Method of a fixed point at frame alignment*

Many methods of image alignment and IGT parameter estimation have small operating areas, i.e. they function only if estimated parameters are small. This section describes a method of estimating IGT parameters with a large operating area. However, the accuracy of this method is low; thus, it can be used to obtain initial parameter approximation, which is then used in a more precise estimation method with a small operating area.

Let mapping $F : W \to W$ of set W into itself be given. A fixed point (FP) of mapping F is any element $w \in W$, for which $F(w) = w$.

That is, while mapping F such a point does not change and transforms into itself. Let's apply the concept of FP to estimate IGT parameters of images [21, 23].

Let's consider two images $x(W)$ and $y(J)$, given on integer m-dimensional grids $W = \{w\} = \{(w_1, \ldots, w_m)\}$ and $J = \{j\} = \{(j_1, \ldots, j_m)\}$. Let the IGT type be known and connect the positions of these images:

$$w = F(j; \bar{\alpha}),$$

where $\bar{\alpha}$ are unknown IGT parameters, which are subject to estimation. After completing an auxiliary transformation IGT P of image $y(J)$, we obtain image $z(J)$, connected with $x(W)$ by means of a complex transformation:

$$w = P(F(j; \bar{\alpha})) = H(j; \bar{\alpha}).$$

Let's suppose that this transformation has only one FP ν:

$$v = H(v; \bar{\alpha}), \tag{6.60}$$

then (6.60) is transformed into a system of m equations relative to parameters $\bar{\alpha}$. If the number of IGT parameters is equal to image dimension, then it's possible to define the parameters from (6.60). If IGT has a larger number of parameters, then it is possible to perform K auxiliary transformation P_k, find FP v_k of each complex transformation H_k, and obtain system of K equations

$$v_k = H_k(v_k; \bar{\alpha}), \quad k = 1, \ldots, K,$$

from which estimates for all transformation F parameters $\bar{\alpha}$ can be obtained.

Figure 6.16 illustrates the place of FP after auxiliary transformation. Figure 6.16(a) shows a smiley. If this image is shifted right on α_1, the smiley will shift on α_1 as well (Figure 6.16(b)). It is obvious that there is no FP at a parallel shift. Let's perform an auxiliary rotation of Figure 6.16(b) for angle π center relative. Then, we obtain an image as in Figure 6.16(c). Figure 6.16(d) shows images in Figures 6.16(a) and 6.16(c) together. It is evident that the point marked by an arrow is the same for both smileys. This is the FP of image transformation of Figure 6.16(a) into Figure 6.16(c).

It is essential that the value of the estimated parameters $\bar{\alpha}$ has no effect on the structure of the proposed algorithm. It is the possibility

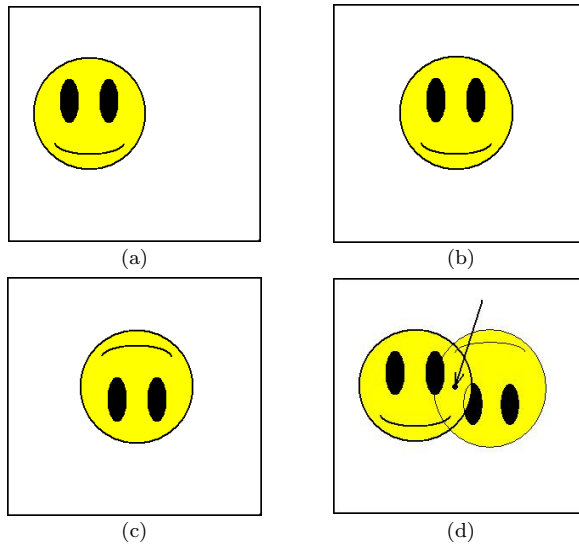


Figure 6.16. Fixed point of a complex transformation shift-rotation.

to find an FP of a complex transformation that matters. For this purpose, it is necessary for the common part of the images $x(W)$ and $z(J)$ not to be too small and contain FP of a complex IGT, i.e. the restrictions for parameter $\bar{\alpha}$ value are not very strong.

It's obvious that the accuracy of parameter $\bar{\alpha}$ estimation depends on the estimation error of an FP position, which has a pixel order, which leads to errors of parameter $\bar{\alpha}$ estimation for the analyzed IGT. In order to make estimation $\bar{\alpha}$ more accurate, it is possible to use the estimate which is obtained with the help of an FP in a more precise algorithm, as an initial approximation. The algorithm can be less time-consuming but with a smaller operating area. Thus, a pair of algorithms allows us to improve the speed of operation in real-time systems, connected with the extraction of useful information from image sequences.

6.5.3.1 *Fixed point method for 2D images*

Let's consider the application of an FP method for estimating IGT parameters of two-dimensional images. The extension of this method on images of larger dimensions leads to technical complications only.

Let $x(u,v)$ and $y(i,j)$ be two 2D images with a known IGT (f,g), which relates the positions of these images:

$$u = f(i,j;\bar{\alpha}), \quad v = g(i,j;\bar{\alpha}), \tag{6.61}$$

where $\bar{\alpha}$ are the estimated parameters. Conducting auxiliary transformation (p,q) of an image $y(i,j)$, we obtain an image $z(i,j)$, connected with $x(u,v)$ by means of a complex transformation:

$$k = f(p(i,j),q(i,j);\bar{\alpha}) = F(i,j;\bar{\alpha}),$$
$$l = g(p(i,j),q(i,j);\bar{\alpha}) = G(i,j;\bar{\alpha}).$$

Let's suppose that this transformation has only one FP (u,v):

$$u = F(u,v;\bar{\alpha}), \quad v = G(u,v;\bar{\alpha}). \tag{6.62}$$

If one manages to find this FP (at least approximately), then (6.62) is transformed into a set of equations relative to parameters $\bar{\alpha}$. If IGT has only two parameters (for example, in the case of a parallel shift), then these parameters can be obtained from (6.62). If IGT has more than two parameters, then it is possible to conduct K auxiliary transformations, find their FP, and obtain a set of equations:

$$u_k = F_k(u_k,v_k;\bar{\alpha}), \quad v_k = G_m(u_k,v_k;\bar{\alpha}), \quad k = 1,\ldots,K,$$

from which estimates of all IGT parameters can be obtained.

The most difficult aspect of this method is to find FP. The necessary criterion for the set of equations (6.61), i.e. (u,v) immobility, is

$$x(F(u,v;\bar{\alpha}),G(u,v;\bar{\alpha})) = z(u,v).$$

But this condition is not sufficient, as there may be other points with values $z(i,j)$ on the image $x(u,v)$, but an FP is only one of them. Therefore, it is necessary to chose an auxiliary transformation for each specific type of IGT and find sufficient characteristics of point immobility.

Figure 6.17 illustrates this method. Cloud-cover images in Figures 6.17(a) and 6.17(b) have IGT parameters (6.62): shift $a = 43.2$, $b = -38.7$, rotation $\alpha = 0.1$ radian, and scale change $s = 0.9$. They are distorted by a strong additive white noise with $\sigma = 30$, so

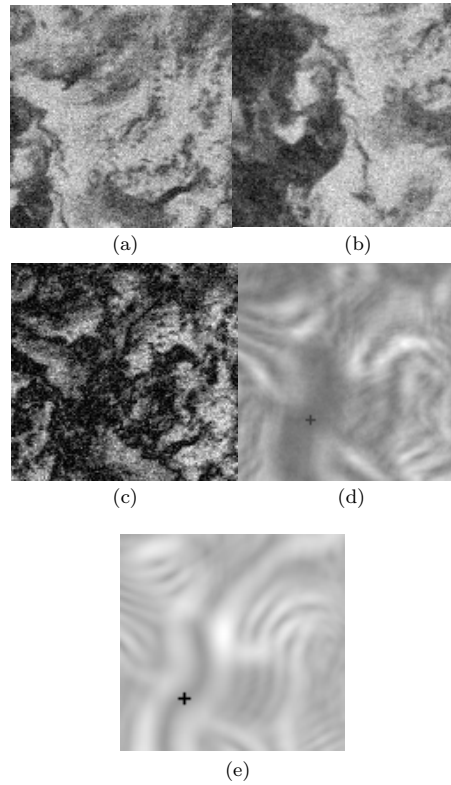


Figure 6.17. Obtaining a fixed point at shift, rotation, scale change, and noise.

the clouds are slightly visible. Figures 6.17(c)–6.17(e) show images $\Delta(i,j)$, $\varepsilon(i,j)$, and $\delta(i,j)$, respectively. Minimum points of images $\varepsilon(i,j)$ and $\delta(i,j)$ are marked with crosses, and their coordinates are (3, −11) and (23, −19). Relative shift estimations, obtained according to statistics $\varepsilon(i,j)$ and $\delta(i,j)$, are respectively (6, −22) and (46, −38). The second one is taken into account. It should be considered as a good result for such rigorous estimation conditions.

The tests of the described algorithm showed that if $|\alpha| \leq 0.1$ radian, $|s-1| \leq 0.1$, and white noise with $\sigma \leq 30$, the estimation error even of large values of shift parameter (tens of pixels) of byte simulated and real images, as a rule, did not exceed 4–6 pixels.

6.6 Adaptive Algorithms of Image Processing

The image processing algorithms discussed above have been synthesized, mainly, assuming that image, noise, and observation models are thoroughly defined. In practice, such models are usually only partially known, i.e. there exists a priory uncertainty in defining an image model. Thus, the synthesis of adaptive processing algorithms is required. There is much literature on adaptive algorithms, in particular, image processing, e.g. [44, 45].

6.6.1 *Pseudo-gradient adaptive algorithms*

According to the purpose of data processing, adaptive algorithms can be divided into two types: argument and criterion. The starting point for algorithm synthesis is minimization of average losses, which is formally expressed by quality functional $R(\bar{\alpha}, Z) = J(\bar{\alpha})$, i.e. the criterion, which value should be minimized according to some parameters $\bar{\alpha}$. However, the requirements for this minimization may be different. The objective of argument tasks is the most accurate determination of the minimum point $\bar{\alpha}^*$ (probably, a variable). This type of task includes measurement, filtration, prediction problems, etc. The objective of criterion tasks is the approximation of $J(\bar{\alpha})$ to its minimal value $J^* = J(\bar{\alpha}^*)$, and the parameters themselves $\bar{\alpha}$ are irrelevant and may greatly differ from $\bar{\alpha}^*$.

According to the method of optimal parameter estimation, adaptive algorithms can be divided into identification and non-identification ones. In identification algorithms first of all, unknown characteristics γ are estimated according to all data available. Then, the estimates $\hat{\gamma}$ obtained are used as the correct ones. As a result, we get the parameters for the algorithms in the form $\bar{\alpha} = \bar{\alpha}(\hat{\gamma})$. This is the core of multiple modified decision rules. The application of these algorithms is complicated by an additional procedure of estimates $\hat{\gamma}$ and instability of further computations to errors of these estimates. In the algorithms without identification, the criterion $J(\bar{\alpha})$ minimization is conducted by variation of $\bar{\alpha}$ without intermediate estimation of any characteristics of the given data. Moreover, $\bar{\alpha}$ may be chosen iteratively during the current processing according to observations over current values $J(\bar{\alpha})$.

To implement such algorithms, it is necessary to estimate current values $J(\bar{\alpha})$, i.e. the criterion should be observable, and this fact

limits the application area of the approach. Sometimes it is possible to replace non-observable $J(\bar{\alpha})$ with the other observable criterion $J_1(\bar{\alpha})$. It is required that minimum points $J(\bar{\alpha})$ and $J(\bar{\alpha})$ coincide in argument tasks, and $J(\bar{\alpha})$ should approximate to $J^* = J(\bar{\alpha}^*)$ as $J(\bar{\alpha})$ approximates to $J_1^* = J_1(\bar{\alpha}_1^*)$ in criterion tasks.

Even if the data is thoroughly described, it is not always possible to find the optimal decision rule because of mathematical difficulties. Even if it is possible to find it, it is often unacceptably time-consuming. Besides, the initial data model usually describes the reality only approximately. For these reasons, it is often impossible to find and apply the optimal rule under real conditions. Therefore, it is necessary to apply the quasi-optimal implementable rules, with a small loss in processing quality. To find such rules, it is possible to use simplified data models, which describe only their fundamental properties. The obtained rules (algorithms) contain some undetermined parameters $\bar{\alpha}$. These rules should be chosen so that the algorithm gives the best result when certain processed data is used.

Let the processing structure be defined and the quality criterion for problem solving be formulated in terms of functional minimization $J(\bar{\alpha})$, which characterizes the average loss when the processing is performed with parameters $\bar{\alpha}$. In view of the unexpected uncertainty of data description, it is impossible to determine the optimal parameters $\bar{\alpha}^*$ beforehand. Therefore, the adaptive procedure is desirable, which together with the processing procedure composes the adaptive algorithm so that parameters $\bar{\alpha}$ are determined in terms of a certain implementation (observation) Z of the object processed.

Thus, the adaptation problem is formulated as a problem of function minimization $J(\bar{\alpha}) = J(\bar{\alpha}, Z)$ for a particular data Z. Pseudogradient (PG) algorithms showed good results while solving this problem. There are some numerical methods for extremum seeking. The most common are various modifications of the gradient algorithm:

$$\bar{\alpha}_n = \bar{\alpha}_{n-1} - \mu_n \nabla J(\bar{\alpha}_{n-1}), \tag{6.63}$$

where $\bar{\alpha}_n$ is approximation to the minimum point of the functional $J(\bar{\alpha})$ following $\bar{\alpha}_{n-1}$, μ_n is a positive numerical sequence determining the step length, and $\nabla J(\bar{\alpha})$ is a gradient of $J(\bar{\alpha})$. Each step in (6.63) is made toward the quickest decrease in $J(\bar{\alpha})$. Although under certain conditions convergence $\bar{\alpha}_n \to \bar{\alpha}^*$ takes place, it may be very slow.

Multiple computations $\nabla J(\bar{\alpha}_{n-1}, Z)$ prevent the application of these methods in image processing. Each computation usually includes the whole processing procedure of Z with parameters $\bar{\alpha}_{n-1}$. It is possible to significantly decrease the amount of computations, if instead of $\nabla J(\bar{\alpha}_{n-1}, Z)$ we take reduction $\nabla Q(\bar{\alpha}_{n-1}) = \nabla J(\bar{\alpha}_{n-1}, Z_n)$, i.e. to calculate the gradient not across the whole Z, but only across some of its part Z_n, for example, in a sliding window. Then in (6.63), instead of an exact gradient value, we use its value with a random error $\bar{\delta}_n$ and obtain

$$\bar{\alpha}_n = \bar{\alpha}_{n-1} - \mu_n(\nabla J(\bar{\alpha}_{n-1}, Z) + \bar{\delta}_n) = \bar{\alpha}_{n-1} - \mu_n \nabla Q(\bar{\alpha}_{n-1}). \quad (6.64)$$

Sequence $\bar{\alpha}_n$ becomes random, so the very fact of its convergence to $\bar{\alpha}^*$ is random as well. Generally speaking, random errors $\bar{\delta}_n$ in (6.64) cannot be a serious obstacle to convergence $\bar{\alpha}_n \to \bar{\alpha}^*$.

The concept of pseudo-gradient (PG) was introduced in [31]. On its basis, a common approach to the analysis and synthesis of algorithms of functional stochastic minimization was developed. The class of PG algorithms is very wide and it includes all (or nearly all) adaptive and learning algorithms. These algorithms are based on the procedure

$$\bar{\alpha}_n = \bar{\alpha}_{n-1} - \mu_n \bar{\beta}_n, \quad (6.65)$$

where $\bar{\beta}_n$ is a random (particularly, deterministic) direction, generally speaking depending on previous values $\bar{\alpha}_i$ and on n.

Vector $\bar{\beta}_n$ is called PG of $J(\bar{\alpha})$ at a point $\bar{\alpha}_{n-1}$, if the pseudo-gradient condition

$$[\nabla J(\bar{\alpha}_{n-1})]^T M[\vec{\beta}_n] \geq 0 \quad (6.66)$$

is fulfilled, where the left-hand member is a scalar product, i.e. on average F is directed at an acute angle to the exact gradient. Algorithm (6.65) is called PG if $\bar{\beta}_n$ is PG at each one-step transition. In this case, transitions in (6.65) will be made on the average toward reduction of $J(\bar{\alpha})$ and it is possible to hope for convergence $\bar{\alpha}_n \to \bar{\alpha}^*$ as $n \to \infty$, although some transitions can be made toward increase in $J(\bar{\alpha})$. Indeed, the fulfillment of relatively weak conditions [31] is sufficient for the convergence with probability one for any initial approximation $\bar{\alpha}_0$.

Note that algorithm (6.65) is much more general than (6.64), as in (6.65) the possibility of calculating $J(\bar{\alpha})$ or $\nabla J(\bar{\alpha})$ is not assumed even with a random error, i.e. $J(\bar{\alpha})$ can be also non-observable. Still the availability of an observable PG is required. In particular, even noisy gradient value of another functional $J_1(\bar{\alpha})$ may be chosen as $\bar{\beta}_n$, but it should have the same minimum point as $J(\bar{\alpha})$.

If we assume that it is possible for $\bar{\beta}_n$ to depend on previous values $\bar{\alpha}_i$, it allows us to use PG algorithms to process not only one-dimensional but also multidimensional data in the order of their scanning.

Until now it was assumed that the task is to find the minimum point $\bar{\alpha}^*$ of the functional $J(\bar{\alpha}, Z)$, which is unique for all Z. Such a point exists, but the processing will be optimal, if data Z is homogeneous. Convergence $\bar{\alpha}_n \to \bar{\alpha}^*$ requires convergence $\mu_n \to 0$. But if we bound μ_n from below (for example, take constants $\mu_n = \mu$), then the dispersions of estimate errors $\bar{\alpha}_n$ of parameters $\bar{\alpha}^*$ will stop decreasing and will have the order μ^2, and $\bar{\alpha}_n$ themselves will oscillate about $\bar{\alpha}^*$. Thus, if homogeneous data processing is conducted simultaneously with the estimation of $\bar{\alpha}^*$ (when $\mu_n = \mu$), then upon reaching steady state operating conditions, only some quasi-optimal processing, i.e. close to optimal, can be conducted.

If data Z is heterogeneous and there is an abrupt change in its characteristics, then optimal parameter values $\bar{\alpha}^*$ can also change. If processing proceeds, then it is possible that immediately after such a rapid change the processing quality will get worse, and then gradually quasi-optimal results will be achieved once again. When there is only a smooth change in observational characteristics Z (more precisely, if optimal parameter values $\bar{\alpha}^*$ change smoothly), which is comparable with the procedure (6.65) transient rate, it is possible to use PG algorithms to process heterogeneous data without their fragmentation into areas of a relatively homogeneous structure.

Thus, PG algorithms are easy to implement and they can be applied to a wide variety of homogeneous and heterogeneous data. Their adaptation can be performed directly during the processing procedure, so data delay line is not required. The indicated positive qualities of PG adaptive algorithms make them preferable in the processing of images and other large data arrays.

Further, we are going to consider the application of PG algorithms for the prediction and image alignment as examples.

6.6.2 *Pseudo-gradient adaptive prediction algorithms*

While solving some tasks of image processing, there often arises an auxiliary problem of their prediction, i.e. the problem of estimation formation $x_{\bar{i}}^*$ of an image element $x_{\bar{i}}$ according to a certain set (pattern) of observations $Z_{\bar{i}}$, which does not include the predicted element itself. In particular, in Kalman filters, the extrapolated estimate is the prediction. In most real situations, the accuracy of prediction increases with the extension of a pattern, but at the same time, the computational costs significantly increase.

The prediction is considered to be optimal, if the minimum of the mean square prediction error $M[(x_{\bar{i}}^* - x_i)^2] = M[\Delta_i^2]$ is achieved. In this case, conditional mean $M[x_{\bar{i}}|Z_{\bar{i}}] = f(Z_{\bar{i}})$, i.e. it is some function of random variables $Z_{\bar{i}}$, is considered to be the optimal prediction. Function f type depends on the distribution type of image elements, i.e. on its model. Thus, the optimal prediction may be presented as $x_{\bar{i}}^* = f(\bar{\alpha}_{\bar{i}}, Z_{\bar{i}})$, where $\bar{\alpha}_{\bar{i}}$ are prediction function parameters. For example, in case of Gaussian images with a zero mean, the optimal prediction is linear: $x_{\bar{i}}^* = \bar{\alpha}_{\bar{i}}^T Z_{\bar{i}}$.

If the form of the prediction function is determined, the problem is reduced to its optimization, i.e. it is necessary to find out the optimal values $\bar{\alpha}_{\bar{i}}^*$ of parameters $\bar{\alpha}_{\bar{i}}$, whereby minimum square error

$$J(\bar{\alpha}_{\bar{i}}) = M[(f(\bar{\alpha}_{\bar{i}}, Z_{\bar{i}}) - x_{\bar{i}})^2] \tag{6.67}$$

is achieved.

Let us form a PG algorithm, which minimizes this functional. It needs an observable pseudo-gradient $\bar{\beta}_n$. Non-observable value $x_{\bar{i}}$ (the problem under consideration is to estimate $x_{\bar{i}}$) is contained in (6.67). Therefore, the functional (6.67) is non-observable, so it is impossible to obtain a PG as an implementation of its gradient. However, observations $z_{\bar{i}} = x_{\bar{i}} + \theta_{\bar{i}}$ are observable; their optimal prediction due to independency and centered of noise coincides with the optimal prediction of elements of a true image. Thus, functional (6.67) can be replaced by an auxiliary functional

$$J_1(\bar{\alpha}_{\bar{i}}) = M[(f(\bar{\alpha}_{\bar{i}}, Z_{\bar{i}}) - z_{\bar{i}})^2] \tag{6.68}$$

its observable PG may be as

$$\bar{\beta}_n = \nabla(f(\bar{\alpha}_n, Z_n) - z_n)^2 = 2\Delta_n \frac{\partial f(\bar{\alpha}_n, Z_n)}{\partial \bar{\alpha}_n}. \tag{6.69}$$

Finding a gradient in (6.69) is rather easy, since the prediction f function is given. For example, if the prediction is linear, then

$$\partial f(\bar{\alpha}_n, Z_n)/\partial \bar{\alpha}_n = \partial f(\bar{\alpha}_n^T Z_n)/\partial \bar{\alpha}_n = Z_n.$$

Note that these algorithms are not associated with any evaluation of image parameters, they are efficient in computation. It enables us to implement them in real-time systems.

Figure 6.18 is an example of PG algorithm application for linear prediction of a homogeneous Gaussian image with a factorable exponential CF. The prediction of an image element is a weighted sum of eight nearest neighbors. Figure 6.18(a) shows the initial image, Figure 6.18(b) shows its optimal prediction, and Figure 6.18(c) represents its errors. Figure 6.18(d) corresponds to the adaptive PG prediction and Figure 6.18(e) represents its errors. Visually, Figures 6.18(b) and 6.18(d) are the same since the adaptive prediction is close to optimal. However, the prediction errors (enlarged for visualization) in Figures 6.18(c) and 6.18(e) are different from each other. In the first few (top) lines of the adaptive prediction (Figure 6.18(d)), errors are relatively large, as the process of prediction parameter adjustment takes place. This process takes place rather quickly. Figure 6.18(f) shows prediction errors during the second image scanning, which develops from the values of prediction parameters established by the end of the first pass. Analysis of Figures 6.18(c) and 6.18(f) allows us to make a conclusion that there is no difference between them. It means that the prediction has nearly converged to an optimal one.

Thus, using the adaptive PG algorithms, it is possible to obtain a good-quality image prediction at a low computational cost.

6.7 Conclusion

This chapter considers a number of new approaches to solving problems of modeling and statistical analysis of multidimensional data sequences, which can be presented in the form of changing images.

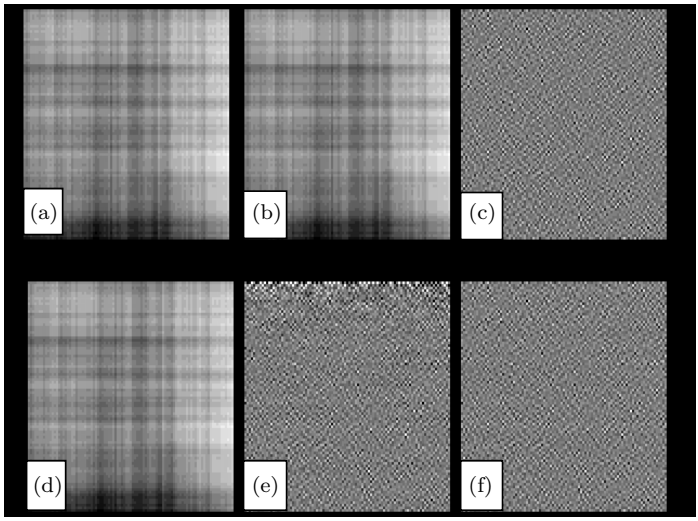


Figure 6.18. Example of an image prediction: (a) initial image; (b) optimal prediction; (c) optimal prediction errors; (d) PG prediction; (e) and (f) PG prediction errors.

First of all, these are problems of mathematical image representations. Taking into consideration image digital format and further computer processing description in the form of a random field, which is set on an integer 2D or multidimensional integer grid, is taken as a basis. The authors analyzed a number of known and new probabilistic models. It enables adequate description of a large class of static and variable images, including ones set on various surfaces. These models are subsequently used to formalize image-processing tasks and synthesize corresponding algorithms.

This chapter touches upon the problems of image filtering, detection of point and lengthy anomalies on the background of noise with spatial correlation, image alignment, and estimation of parameters of their interframe geometric transformations. Optimal and quasi-optimal algorithms, which can help solve these problems, were obtained. Their efficiency and computational complexity were analyzed. Special attention is paid to adaptive algorithms of image processing when there is uncertainty in the description of the processed data.

Acknowledgements

This chapter was supported by grant No. 23-21-00249 from the Russian Science Foundation.

References

[1] Bondur, V.G. (2003) Modeling of multispectral aerospace image brightness dynamic fields. *Study of Earth from Space*, 2, 3–17.

[2] Box, G.E.P., Jenkins, G.M., and Reinsel, G.C. (2008) *Time Series Analysis: Forecasting and Control*, 4th Ed. (Wiley), 756 p.

[3] Dikshit, S.S. (1984) A recursive Kalman window approach to image restoration. *IEEE Transactions*, 32, 125–139.

[4] Duda, R.O., Hart, P.E., and Stork, D.G. (2000) *Pattern Classification*, 2nd Ed. (Wiley-Interscience, New York).

[5] Dudgeon, D.E. (1995) *Multidimensional Digital Signal Processing* (Prentice Hall), 406 p.

[6] Franke, B., Plante, J.-F., Roscher, R., Lee, A., Smyth, C., Hatefi, A., Chen, F., Gil, E., Schwing, A., Selvitella, A., Hoffman, M.M., Grosse, R., Hendricks, D., and Reid, N. (2016) Statistical inference, learning and models in big data. *International Statistical Review*, 84(3), 371–389. doi:10.1111/insr.12176.

[7] Favorskaya, M.N. and Levtin, K. (2014) Early smoke detection in outdoor space by spatio-temporal clustering using a single video camera. Advances in Intelligent Systems and Computing, Vol 234, pp. 43–56.

[8] Gimel'farb, G.L. (1999) *Image Textures and Gibbs Random Fields* (Springer), 263 p.

[9] Gonzalez, R. and Woods, R. (2008) *Digital Image Processing*, 3rd Ed. (Pearson), p. 647.

[10] Gruzman, I.C., Kirichuk, V.P., Kosikh, G.I., Peretryagin, G.I., and Spector, A.A. (2000) Digital image processing in informative systems. Novosibirsk State Technical University, 168 p. (in Russian).

[11] Habibi, A. (1972) Two dimensional Bayesian estimate of images. *Proceedings of IEEE*, 30(7), 878–883.

[12] Hafizov, R.G., Okhotnikov, S.A., and Yarantsev, T.V. (2016) Model of image edges of objects in presence of geometric distortions. *Computer Optics*, 40(3), 404–409.

[13] Huang, T.S. (ed.) (1981) *Image Sequence Analysis* (Springer, Berlin), 452 p.

[14] Huang, T.S. (ed.) (1983) *Image Sequence Processing and Dynamic Scene Analysis* (Springer-Verlag, New York).
[15] Jain, L.C. and Favorskaya, M.N. (2015) Practical matters in computer vision. Intelligent Systems Reference Library, Vol. 75.
[16] Jähne, B. (2005) *Digital Image Processing*, 6th Ed. (Springer, Berlin).
[17] Kozlovsky, A.N. (2016) Mathematical model basic and simple of the object image based on contour analysis. *International Scientific Journal*, 3, 3–7.
[18] Krasheninnikov, V.R. (2011) Wave image models on the surfaces. *Proceedings of 8th Open German-Russian Workshop* "*Pattern Recognition and Image Understanding*", Nizhny Novgorod, pp. 154–157.
[19] Krasheninnikov, V.R., Kalinov, D.V., and Pankratov, Y.G. (2001) Spiral autoregressive model of a quasi periodic signal. *Pattern Recognition and Image Analysis*, 8(1), 211–213.
[20] Krasheninnikov, V.R. (2015) Correlation analysis and synthesis of random field wave models. *Pattern Recognition and Image Analysis*, 25(1), 41–46.
[21] Krasheninnikov, V.R. and Potapov, M.A. (2011) A way to detect the straight line trajectory of an immovable point for estimating parameters of geometrical transformation of 3D images. *Pattern Recognition and Image Analysis*, 21(2), 280–284.
[22] Krasheninnikov, V.R., Armer, A.I., Kuznetsov, V.V., and Lebedeva, E.Y. (2011) Cross-correlation portraits of voice signals in the problem of recognizing voice commands according to patterns. *Pattern Recognition and Image Analysis*, 21(2), 192–194.
[23] Krasheninnikov, V.R. and Potapov, M.A. (2012) Estimation of parameters of geometric transformation of images by fixed point method. *Pattern Recognition and Image Analysis*, 22(2), 303–317.
[24] Krasheninnikov, V.R., Kuznetsov, V.V., Lebedeva, E.Y., and Krasheninnikova, N.A. (2013) Optimization of dictionary and model library for recognition of speech commands based on cross-correlation portraits. *Pattern Recognition and Image Analysis*, 23(1), 80–86.
[25] Krasheninnikov, V.R. and Kopylova, A.S. (2012) Algorithms for automated processing images of blood serum facies. *Pattern Recognition and Image Analysis*, 22(4), 583–592.
[26] Landgrebe, D.A. (2003) *Signal Theory Methods in Multispectral Remote Sensing* (John Wiley), p. 520.
[27] Lorenzi, M., Simpson, I.J., Mendelson, A.F., Vos, S.B., Cardoso, M.J., Modat, M., Schott, J.M., and Ourselin, S. (2016) Multimodal image analysis in Alzheimer's disease via statistical modelling of non-local intensity correlations. *Scientific Reports*, 6, 8 p. Article number: 22161. doi:10.1038/srep22161.

[28] Oord, A., Kalchbrenner, N., Vinyals, O., Espeholt, L., Graves, A., and Kavukcuoglu, K. (2016) Conditional image generation with PixelCNN decoders. *European Conference on Computer Vision*, 12 p.

[29] Polyak, B.T. and Tsypkin, Y.Z. (1980) Optimal pseudogradient adaptation procedure. *Aautomation and Remote Control*, 41(8), 1101.

[30] Pratt, W.K. (2001) *Digital Image Processing. PIKS Inside*, 3rd edn. (Wiley, New York).

[31] Prewitt, J.M.S. (1970) Object enhancement and extraction. In Lipkin, B.S. and Rosenfeld, A. (eds.) *Picture Processing and Psychopictorics* (Academic Press, New York).

[32] Sergeev, V.V. and Denisov, A.Y. (2013) The iterative method for reconstruction of piecewise constant images with known region boundaries. *Computer Optics*, 37(2), 239–243.

[33] Sergeev, V.V. and Usikov, V.V. (2016) The parametric model of the autocorrelation function of space hyperspectral images. *Computer Optics*, T40(3), 416–421.

[34] Serra, J. (1988) *Image Analysis and Mathematical Morphology. Theoretical Advances* (Academic Press, London).

[35] Siwei, L. and Eero, P.S. (2009) Nonlinear extraction of 'independent components' of natural images using radial Gaussianization. *Neural Computation*, 18(6), 1–35.

[36] Smelik, N.D. and Filchenkov, A.A. (2016) Multimodal topic model for texts and images utilizing their embeddings. *Machine Learning and Data Analysis*, 2(4), 421–441.

[37] Shalygin, A.S. and Palagin, Y.I. (1986) *Applied Methods of Statistical Modelling* (Mashinostroenie, Leningrad).

[38] Soifer, V.A., Popov, S.B., Myasnikov, V.V., and Sergeev, V.V. (2009) *Computer Image Processing. Part I. Basic Concepts and Theory* (VBM Verlag Dr. Müller), 300 p.

[39] Soifer, V.A. (2015) *Promising Information Technologies Remote Sensing* (Samara), p. 237.

[40] Vasil'ev, K.K. (1999) Statistical analysis of multidimensional images. *Pattern Recognition and Image Analysis*, 9(4), 732–748.

[41] Vasil'ev, K.K., Dement'ev, V.E., and Andriyanov, N.A. (2015) Doubly stochastic models of images. *Pattern Recognition and Image Analysis*, 25(1), 105–110.

[42] Vasil'ev, K.K. and Popov, O.V. (1999) Autoregression models of random fields with multiple roots. *Pattern Recognition and Image Analysis*, 9(2), 327–328.

[43] Vasil'ev, K.K. and Krasheninnikov, V.R. (2017) *Statistical Analysis of Images Sequences* (Radiotehnika, Moscow), 248 p. (in Russian).

[44] Vasil'ev, K.K., Dement'ev, V.E., and Luchkov, N.V. (2012) Analysis of efficiency of detecting extended signals on multidimensional greeds. *Pattern Recognition and Image Analysis*, 22(2), 400–408.

[45] Vasiliev, K.K. and Luchkov, N.V. (2014) Imitazione immagini disomogeneo. *Italian Science Review*, 5(14), 74–77.

[46] Vasil'ev, K.K., Dement'ev, V.E., and Andriyanov, N.A. (2016) Application of mixed models for solving the problem on restoring and estimating image parameters. *Pattern Recognition and Image Analysis*, 26(1), 240–247.

[47] Vizilter, Y.V., Pyt'Ev, Y.P., Chulichkov, A.I., and Mestetskiy, L.M. (2015) Morphological image analysis for computer vision applications. *Intelligent Systems Reference Library*, Vol. 73.

[48] Widrow, B. and Stearns, S.D. (1985) *Adaptive Signal Processing* (Prentice-Hall Inc., Englewood Cliffs).

[49] Woods, J.W. (1981) Two-dimensional Kalman filtering. *Topics in Applied Physics*, Vol. 42 (Berlin), pp. 155–208.

[50] Xiang-Yang Wang, Lin-Lin Liang, Yong-Wei Li, and Hong-Ying Yang (2017) Image retrieval based on exponent moments descriptor and localized angular phase histogram. *Multimedia Tools and Applications*, 76(6), 7633–7659.

[51] Yaroslavsky, L. (1985) *Digital Picture Processing. An Introduction* (Springer, Berlin).

[52] Yan, X., Yang, J., Sohn, K., and Lee, H. (2016) Attribute2Image: Conditional image generation from visual attributes. *The 14th European Conference on Computer Vision.* Accepted by ECCV 2016, 19 p.

[53] Zhuravlev Yu, I. (1998) An algebraic approach to recognition or classifications problems. *Pattern Recognition and Image Analysis*, 8(1), 59–100.

© 2025 World Scientific Publishing Company
https://doi.org/10.1142/9789811267215_0007

Chapter 7

Conjugation Index as a Proximity Measure in Problems of Recognition

Vladimir A. Fursov[*,‡], **Sergey A. Bibikov**[†,§], **and Denis A. Zherdev**[†,¶]

[*]*Research Education Center on the Computer Science, Samara National Research University, Samara, the Russian Federation*
[†]*Supercomputers and Computer Science Department, Samara National Research University, Samara, the Russian Federation*
[‡]*fursov@ssau.ru*
[§]*bibikov.sergei@gmail.com*
[¶]*denis.zherdev.91@mail.ru*

This chapter is devoted to problems of recognition in situation when feature space dimension is significantly higher than a number of available training vectors. A typical example is the case when feature vectors are made up of image pixels. The object of our exploration is the so-called conjugation index used as a proximity measure of a vector to a class. This class is vector space formed by a set of training vectors. The decision on vector belonging to some class is made if the conjugation indexes with this class are maximum. We suggest the criterion of the class separability by the conjugation index. With the use of this criterion, we form so-called support subspaces from the training vectors that can cut computing complexity and increase the quality of the pattern recognition. This chapter gives application examples in the following problems: face recognition, thematic classification of hyperspectral images, recognition of synthetic aperture radar images, etc.

Keywords: Support vector machine, k-nearest neighbors, Manhattan metric, conjugation index, support subspaces, face recognition, neural network, principal component analysis, thematic classification, hyperspectral images, synthetic aperture radar, moving and stationary target acquisition and recognition, Indian Pines.

7.1 Introduction

It is difficult to find areas where recognition is not used. Especially, the systems of image recognition and understanding are used widely. The reason for this is the fact that information about many objects or scenes is registered and stored in the form of digital images. The classification methods based on the computation of the proximity measures between images were among the first approaches to the pattern recognition problem solution. The experimental studies of the recognition methods based on the proximity idea confirm its efficiency.

The Euclidean distance (the Euclidean metric) is most widely used in this direction. The Euclidean metric is the "ordinary" straight-line distance between two points in Euclidean space, which is finite-dimensional a vector space with the positive definite scalar product generating norm of L2 in it. The Euclidean metrics is a base for designing of huge number of recognition and classification methods. We notice that with the use of this metric many popular methods such as support vector machine (SVM), the algorithm of the k-nearest neighbors (k-NN), etc. are fulfilled. Along with the Euclidean metric, also the Makhalanobis distance and the Manhattan metric are widely used.

Comparative research shows that recognition quality can significantly differ with use of different proximity measures. These distinctions depend also on the specific of the problem. Naturally, there is a question of the proximity measure application, which in certain conditions can yield the best result.

Therefore, along with the creation of recognition methods on the basis of the widely known proximity measures, new proximity measures are offered as well. For example, in [31] a new diagnostic model and set of a decision rule based on the combination of two-dimensional vectors of independent function are offered. In [5], the authors propose new measures of proximity for algorithms of the multidepot vehicle routing problem with time windows. It is shown

that the inclusion of time windows in proximity measures gives better results than only using the distance, in terms of routing problems.

In this chapter, we introduce the results of research, which prove the possibility of a use for recognition and classification so-called index of conjugacy [16, 20] with space (or zero space) created by training vectors of a recognized class (Conjugation Index Method — CIM). We note that decision rules are constructed on the basis of the conjugation index in a feature space, which forms from values of brightness sample, i.e. of the digital image pixels.

The important feature of the offered method is that it characterizes closeness with a plurality of vectors of each class. It appears that it must lead to a fuller use of all available information on a class in each specific situation. The specified circumstance served as a motive for carrying out comprehensive investigations of conjugation index efficiency as a proximity measure in recognition problems. Examples of the solution for some application-oriented tasks are presented in this chapter.

7.2 Recognition Problem Formulation Using the Conjugation Index

The mathematical formulation of the image recognition problem is a standard problem of pattern recognition. Let M be a number of available images of the same size for some object. Each image can be transformed to $N \times 1$-vector $\mathbf{x} = [x_1, x_2, \ldots, x_N]^T$, where $x_1, x_2, \ldots, x_N$ are components of the feature vector and N is the total number of pixels/features. We suppose the set of feature vectors corresponding to images of one object forms the class. These M vectors can be used as a training dataset in the form of $N \times M$-matrix of the class:

$$\mathbf{X} = [\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_M]. \tag{7.1}$$

Suppose, K of the various objects and M_K of training vectors for each object are given (for simplicity, we assume that equal number M of training vectors is given for each object). Thus, matrices $\mathbf{X}_k, k = \overline{1, K}$ can be formed for K classes.

Let $\mathbf{x}$ be the feature vector of an object from an unknown class. The problem of the classifier creation consists to define of decision function $f(\mathbf{x})$ for determination of a class number to which belongs

vector $\mathbf{x}$. Let $\mathbf{X}_k$ be $N \times M$-matrix of a kth class (7.1). We define a proximity measure of a vector $\mathbf{x}$ to kth class as a conjugation index of $\mathbf{x}$ with the subspace formed by vectors of the matrix $\mathbf{X}_k$:

$$R_k = (\mathbf{x}^T \mathbf{X}_k [\mathbf{X}_k^T \mathbf{X}_k]^{-1} \mathbf{X}_k^T \mathbf{x})(\mathbf{x}^T \mathbf{x})^{-1}. \tag{7.2}$$

Along with a conjugation index (7.2), we consider a conjugation index with zero space of the matrix $\mathbf{X}_k$:

$$S_k = (\mathbf{x}^T \mathbf{T}_{k,0} \mathbf{T}_{k,0}^T \mathbf{x})(\mathbf{x}^T \mathbf{x})^{-1}. \tag{7.3}$$

Here $\mathbf{T}_{k,0}$ is the $N \times (N - M)$-matrix made of eigenvectors corresponding to zero eigenvalues of an $N \times N$-matrix $\mathbf{X}_k \mathbf{X}_k^T$.

Now, we formulate the rule for the use of conjugation indexes (7.2) and (7.3) as proximity measures for decision function creation. We assume that for each kth class one of the following $N \times N$-matrixes is created:

$$\mathbf{Q}_{k,R} = \mathbf{X}_k [\mathbf{X}_k^T \mathbf{X}_k]^{-1} \mathbf{X}_k^T \tag{7.4}$$

or

$$\mathbf{Q}_{k,S} = \mathbf{T}_{k,0} \mathbf{T}_{k,0}^T. \tag{7.5}$$

Then decision function $f(\mathbf{x})$ can be constructed as follows. The vector $\mathbf{x}$ belongs to mth class, i.e.

$$f(\mathbf{x}) = m, \quad m = 1, 2, \ldots K,$$

if

$$R_m = \max_k R_k, \quad \text{where } R_k = (\mathbf{x}^T \mathbf{Q}_{k,R} \mathbf{x})(\mathbf{x}^T \mathbf{x})^{-1} \tag{7.6}$$

or

$$S_m = \min_m S_k, \quad \text{where } S_k = (\mathbf{x}^T \mathbf{Q}_{k,S} \mathbf{x})(\mathbf{x}^T \mathbf{x})^{-1}. \tag{7.7}$$

It should be noted that the use of the conjugation index (7.2) as a proximity measure is possible only in the case when the training vectors are independent, and their dimension exceeds the number of the training vectors in each class since the matrix $\mathbf{X}_k^T \mathbf{X}_k$ must be nonexceptional. Otherwise, it is necessary to carry out the selection of some number of the training vectors. At the same time, it is necessary

to answer two questions: how many vectors should be included in matrixes $\mathbf{X}_k$ and what is the selection rule?

Algorithms of training vector selection can be various. The goal of these algorithms is to achieve the maximum recognition quality. Let us denote as *support subspaces* the subspaces formed by the training sets of each class which are selected with respect to the specified requirements. We discuss the support subspace formation procedures in Section 7.5.

7.3 Geometrical Interpretation

It could be interesting to give the geometrical interpretation of the conjugation indexes. At first, we consider the conjugation index determined by formula (7.2). We use an approach based on orthogonal transformations of the $\mathbf{Q}_{k,R}$ matrix (7.4).

Formula (7.4) of matrix $\mathbf{Q}_{k,R}$ (hereinafter matrices have no indexes, for simplicity) can be rewritten as

$$\mathbf{Q}_{k,R} = \mathbf{X}\mathbf{F}\mathbf{F}^T[\mathbf{X}^T\mathbf{X}]^{-1}\mathbf{F}\mathbf{F}^T\mathbf{X}^T, \tag{7.8}$$

where $\mathbf{F}$ is the $M \times M$–matrix such that

$$\mathbf{X}^T\mathbf{X} = \mathbf{F}\mathbf{\Lambda}\mathbf{F}^T,$$

diagonal matrix $\mathbf{\Lambda} = \mathrm{diag}(\lambda_1, \lambda_1, \ldots, \lambda_M)$ is composed of M eigenvalues $\lambda_i > 0, i = \overline{1, M}$, and columns of matrix $\mathbf{F}$ are corresponding eigenvectors. Representation (7.8) is equivalent to (7.4) because of the following equations:

$$\mathbf{F}\mathbf{F}^T = \mathbf{F}^T\mathbf{F} = \mathbf{E}, \quad \mathbf{F}^{-1} = \mathbf{F}^T, \quad [\mathbf{F}^T]^T = \mathbf{F},$$
$$[\mathbf{A}\mathbf{B}\mathbf{C}]^{-1} = \mathbf{C}^{-1}\mathbf{B}^{-1}\mathbf{A}^{-1},$$

and

$$\mathbf{F}^T[\mathbf{X}^T\mathbf{X}]^{-1}\mathbf{F} = \mathbf{F}^{-1}[\mathbf{X}^T\mathbf{X}]^{-1}[\mathbf{F}^T]^{-1} = [\mathbf{F}^T\mathbf{X}^T\mathbf{X}\mathbf{F}]^{-1} = \mathbf{\Lambda}^{-1}. \tag{7.9}$$

Formula (7.8) can be rewritten by using equalities (7.9) as follows:

$$\mathbf{Q}_{k,R} = \mathbf{X}\mathbf{F}\mathbf{\Lambda}^{-1}\mathbf{F}^T\mathbf{X}^T = \mathbf{X}\mathbf{F}\mathbf{\Lambda}^{-1/2}\mathbf{F}^T\mathbf{X}^T. \tag{7.10}$$

Let us consider a matrix

$$\mathbf{T}_\lambda = \mathbf{X}\mathbf{F}\mathbf{\Lambda}^{-1/2}. \tag{7.11}$$

It is possible to show that columns $\mathbf{t}_{\lambda,i}, i = \overline{1,M}$ of matrixes $\mathbf{T}_\lambda$ are eigenvectors corresponding to non-zero eigenvalues of the matrix $\mathbf{X}\mathbf{X}^T$. With the use of equality (7.11), formula (7.10) can be rewritten as

$$\mathbf{Q}_{k,R} = \mathbf{T}_\lambda \mathbf{T}_\lambda^T. \tag{7.12}$$

In (7.11), we can use the normalized vector $\bar{\mathbf{x}} : \|\bar{\mathbf{x}}\| = 1$ instead of $\mathbf{x}$ to neglect the denominator $\mathbf{x}^T\mathbf{x}$. Thus, in accordance with formula (7.12), the conjugation index (7.2) for vector $\bar{\mathbf{x}}$ can be presented as

$$R(\bar{\mathbf{x}}) = \mathbf{V}_\lambda^T \mathbf{V}_\lambda, \tag{7.13}$$

where $\mathbf{V}_\lambda$ is the $M \times 1$-vector:

$$\mathbf{V}_\lambda = \mathbf{T}_\lambda^T \bar{\mathbf{x}} = \left[\cos\left(\widehat{\mathbf{t}_{\lambda,1}, \mathbf{x}}\right), \cos\left(\widehat{\mathbf{t}_{\lambda,2}, \mathbf{x}}\right), \ldots, \cos\left(\widehat{\mathbf{t}_{\lambda,M}, \mathbf{x}}\right)\right]^T. \tag{7.14}$$

Taking into account (7.13) and (7.14), we can write

$$R(\mathbf{x}) = \sum_{i=1}^{M} \cos^2\left(\widehat{\mathbf{t}_{\lambda,i}, \mathbf{x}}\right). \tag{7.15}$$

Thus, the conjugation index (7.2) is the sum of the squared cosines of the angles between a vector $\mathbf{x}$ and eigenvectors $\mathbf{t}_{\lambda,i}$ corresponding to non-zero eigenvalues of the matrix $\mathbf{X}\mathbf{X}^T$.

Using (7.5) and (7.7), it is easy to deduce similar representation for the conjugation index $S_k(\mathbf{x})$ of the vector $\mathbf{x}$ with zero space of the matrix $\mathbf{X}$. Unlike formula (7.15), we use eigenvectors $\mathbf{t}_{0,i}, i = \overline{1, N-M}$ corresponding to zero eigenvalues of the matrix $\mathbf{X}\mathbf{X}^T$ instead of $\mathbf{t}_{\lambda,i}$:

$$S(\mathbf{x}) = \sum_{i=1}^{N-M} \cos^2(\widehat{\mathbf{t}_{0,i}, \mathbf{x}}). \tag{7.16}$$

In conclusion, it should be noted that conjugation indexes (7.2) and (7.3) relate as follows:

$$R(\mathbf{x}) = 1 - S(\mathbf{x}). \tag{7.17}$$

Let us prove it. First, we are going to show the correctness of equality

$$\mathbf{E}_N - \mathbf{X}[\mathbf{X}^T\mathbf{X}]^{-1}\mathbf{X}_k^T = \mathbf{T}_0\mathbf{T}_0^T, \tag{7.18}$$

where $\mathbf{E}_N$ is the unity $N \times N$-matrix and $\mathbf{T}_0$ is the same, as a matrix $\mathbf{T}_{k,0}$ in (7.3). Equality (7.17) can be proven by the multiplication of both parts on the left by $\mathbf{T}_0^T$ and on the right by $\mathbf{T}_0$ with the use of properties

$$\mathbf{T}_0^T\mathbf{X} = 0 \quad (\mathbf{X}^T\mathbf{T}_0 = 0), \quad \mathbf{T}_0^T\mathbf{T}_0 = \mathbf{E}_{N-M}. \tag{7.19}$$

Further, according to definition (7.2), it is possible to write down

$$\begin{aligned} 1 - R(\mathbf{x}) &= \frac{\mathbf{x}^T\mathbf{x}}{\mathbf{x}^T\mathbf{x}} - \frac{\mathbf{x}^T\mathbf{X}[\mathbf{X}^T\mathbf{X}]^{-1}\mathbf{X}^T\mathbf{x}}{\mathbf{x}^T\mathbf{x}} \\ &= \frac{\mathbf{x}^T[\mathbf{E}_N - \mathbf{X}[\mathbf{X}^T\mathbf{X}]^{-1}\mathbf{X}^T]\mathbf{x}}{\mathbf{x}^T\mathbf{x}}. \end{aligned} \tag{7.20}$$

At last, inserting $\mathbf{T}_0\mathbf{T}_0^T$ from (7.19) to the right part (7.20) instead of expression in square brackets, in accordance with (7.18), we obtain equality (7.17). Thus, indexes (7.2) and (7.3) for a given $\mathbf{x}$ and $\mathbf{X}$ are equivalent and interchangeable (see (7.17)). Therefore, using particularly one of two conjugation indexes (7.2) or (7.3) can be determined only by the desire to reduce calculations. Further, we use everywhere mostly a conjugation index (7.2).

7.4 Separability by the Conjugation Index and Support Subspace Form

We introduced the concept of *support subspaces* in (7.2). In this section, we discuss a method of supporting subspace formation by the example of recognition of two classes. We assume that for two objects, the image sets were received approximately in identical conditions. Nonetheless, these obtained images have some differences due to the influence of random noise, different points of view, etc. Our purpose is to create support subspaces for two classes, k and l, which are represented by $N \times M$-matrices $\mathbf{X}_k$ and $\mathbf{X}_l$.

Our task consists of the development of the formation rule for the matrixes $\mathbf{X}_k$ and $\mathbf{X}_l$ (support subspaces). The rule is used for a binary classification problem solution. The problem is in defining

the vector $\mathbf{x}$ to belong to one of the two classes k or l with the use of expressions (7.2) and (7.6).

We introduce the term of the *separability by the conjugation index*. Let us give the definition: the classes k and l are separable by the conjugation index if any vectors within these classes:

$$\forall i = \overline{1, M}, j = \overline{1, M} : \mathbf{x}_{k,i} \in \mathbf{X}_k, \mathbf{x}_{l,j} \in \mathbf{X}_l, \tag{7.21}$$

following conditions satisfy:

$$R(\mathbf{x}_{k,i}, \mathbf{Q}_k) > R(\mathbf{x}_{k,i}, \mathbf{Q}_l), \tag{7.22}$$

$$R(\mathbf{x}_{l,j}, \mathbf{Q}_l) > R(\mathbf{x}_{l,j}, \mathbf{Q}_k). \tag{7.23}$$

Here $R(\mathbf{x}_{k,i}, \mathbf{Q}_k), R(\mathbf{x}_{k,i}, \mathbf{Q}_l)$ are the conjugation indices of the vector $\mathbf{x}_{k,i}$ with its own k-class and with opposite l-class respectively, $R(\mathbf{x}_{l,j}, \mathbf{Q}_k), R(\mathbf{x}_{l,j}, \mathbf{Q}_l)$ are the conjugation indices of the vector $\mathbf{x}_{l,j}$ from lth class with classes k and l, which are connected with $N \times N$-matrixes $\mathbf{Q}_k, \mathbf{Q}_l$ respectively.

Both classes are correctly recognized by the decision rule (7.6), if the expressions (7.22) and (7.23) are satisfied. Unfortunately, the specified inequalities (7.22) and (7.23) aren't always carried out. To overcome this difficulty, it is necessary to construct such support subspaces, for which these conditions are provided. Therefore, the question is how to obtain such separable support subspaces. To answer this question, we consider the following procedure.

We suppose that each of two classes contains unique vectors. Thus, corresponding $N \times M$-matrixes $\mathbf{X}_k$ and $\mathbf{X}_l$ do not contain identical columns. We can choose two column vectors:

$$\mathbf{x}_{k,I} \in \mathbf{X}_k, \quad \mathbf{x}_{l,J} \in \mathbf{X}_l, \tag{7.24}$$

which satisfy to following criteria:

$$R(\mathbf{x}_{k,I}, \mathbf{Q}_{l,J}) = \max_{i,j=1,M} R(\mathbf{x}_{k,i}, \mathbf{Q}_{l,j}), \tag{7.25}$$

$$R(\mathbf{x}_{l,J}, \mathbf{Q}_{k,I}) = \max_{i,j=1,M} R(\mathbf{x}_{l,j}, \mathbf{Q}_{k,i}), \tag{7.26}$$

where $\mathbf{Q}_{l,j}$ and $\mathbf{Q}_{k,i}$ are formed from matrixes $\mathbf{X}_l$ and $\mathbf{X}_k$ by excluding $\mathbf{x}_{l,j}$ and $\mathbf{x}_{k,i}$ correspondingly.

It is evident that if conditions (7.22) and (7.23) are satisfied for vectors $\mathbf{x}_{k,I}, \mathbf{x}_{l,J}$ and matrixes $\mathbf{Q}_{k,I}, \mathbf{Q}_{l,J}$ obtained using (7.4), then

conditions (7.22) and (7.23) are satisfied for other vectors. Therefore, obtained support subspaces $\mathbf{X}_k$ and $\mathbf{X}_l$ of size $N \times M-1$ are separable by the conjugation index. These support subspaces can be used for recognition with the use of matrices $\mathbf{Q}_{k,I}$ and $\mathbf{Q}_{l,J}$.

If both or at least one of conditions (7.22) and/or (7.23) are not satisfied, we exclude both vectors (or one of them) from the corresponding matrices $\mathbf{X}_k$ and $\mathbf{X}_l$. After that conditions (7.22) and (7.23) are checked for newly formed reduced matrixes $\mathbf{X}_k$ and $\mathbf{X}_l$. The sizes of matrixes $\mathbf{X}_k$ and $\mathbf{X}_l$ at this step can be reduced to $N \times M - 1$ or be same $N \times M$. If conditions (7.22) and (7.23) are satisfied, we use decisive matrixes $\mathbf{Q}_{k,I}$ and $\mathbf{Q}_{l,J}$ for recognition. Otherwise, we continue the process of iterative vector exclusion from matrices as well, as described above. The process of vector exception continues until we meet both conditions (7.22) and (7.23).

It should be noted that the obtained support subspaces can differ in size. In particular, as a result of the exclusion procedure, each class can contain only one vector. Therefore, it is always possible to construct the support subspaces (including unidimensional), which satisfy conditions (7.22) and (7.23). In the case of both support subspaces being unidimensional, the decisive rule (7.6) is a well-known criterion of a spectral angle [19, 28].

The aspiration to provide strict satisfying of the separability conditions (7.22), (7.23) can lead to an exclusion of a big number of vectors, and as a result, to loss of essential information on a class. Higher quality of recognition is usually reached by a compromise between separability conditions (7.22) and (7.23) and informativity of a support subspace. Such compromise is inevitable for many methods of recognition. For example, for a method of SVM (*Support Vector Machine*), compromise is reached by penalty function use in the case when classes linear separability conditions aren't satisfied. We postpone research on support subspace formation procedures' influence on recognition quality until Section 7.10, in which examples of radar image recognition are given.

It is necessary to draw attention to the following problem. In order to realize the vectors exception algorithm, the matrices $\mathbf{X}_k$ and $\mathbf{X}_l$ must be non-singular. Therefore, detection and removal of linear dependent vectors is usually the initial stage of a support subspace formation algorithm. It is easy to carry out by calculation of a correlation coefficient for all possible vector couples from each class. One must exclude vector from vectors couple, for which the correlation

coefficient is the greatest. The vectors exclusion procedure repeats until the prescribed vector quantity or acceptable threshold value is achieved.

During the binary classification, we perform the separation of each pair of obtained support subspaces. However, we can recognize a bigger number of classes. In this case, current recognizable vector belongs to the class, which receives the highest number of votes in the one-vs-one. This approach to classes recognition we used in [11].

7.5 Clustering by Conjugation Index

Decrease in computing complexity and improvement in classification quality can be reached by splitting a class into subclasses. The conjugation index is convenient for that purpose. The procedure is implemented as follows. In the first step, from the M vectors of a particular class, we choose a pair of the most "remote" vectors (let us denote them as $\mathbf{x}_1, \mathbf{x}_M$), for which the value of the correlation

$$R_{1,M} = \langle \mathbf{x}_1^T \mathbf{x}_M \rangle / \|\mathbf{x}_1\| \|\mathbf{x}_M\| \tag{7.27}$$

achieves its minimum. Then from the remaining $M-2$ vectors, we find vectors $\mathbf{x}_2, \mathbf{x}_{M-1}$, for which corresponding correlation coefficients

$$R_{1,2} = \langle \mathbf{x}_1^T \mathbf{x}_2 \rangle / \|\mathbf{x}_1\| \|\mathbf{x}_2\|, \tag{7.28}$$

$$R_{M-1,M} = \langle \mathbf{x}_{M-1}^T \mathbf{x}_M \rangle / \|\mathbf{x}_{M-1}\| \|\mathbf{x}_M\| \tag{7.29}$$

reach their maximum values. The vector couples $\mathbf{x}_1, \mathbf{x}_2$ and $\mathbf{x}_{M-1}, \mathbf{x}_M$ generate two subspaces, which are described by matrices $\mathbf{X}_{1,2}$ and $\mathbf{X}_{M-1,M}$ respectively. In the following step, the next two vectors $\mathbf{x}_3, \mathbf{x}_{M-2}$ are determined. These vectors are chosen to be the closest to constructed subspaces by the criterion of the maximum conjugation indexes:

$$R_{1,2,3} = (\mathbf{x}_3^T \mathbf{X}_{1,2} [\mathbf{X}_{1,2}^T \mathbf{X}_{1,2}]^{-1} \mathbf{X}_{1,2}^T \mathbf{x}_3)(\mathbf{x}_3^T \mathbf{x}_3)^{-1}, \tag{7.30}$$

$$R_{M-2,M-1,M} = \frac{(\mathbf{x}_{M-2}^T \mathbf{X}_{M-1,M} [\mathbf{X}_{M-1,M}^T \mathbf{X}_{M-1,M}]^{-1} \mathbf{X}_{M-1,M}^T \mathbf{x}_{M-2})}{(\mathbf{x}_{M-2}^T \mathbf{x}_{M-2})}. \tag{7.31}$$

As a result, on a step with number $M/2$ (for even M) or $(M-1)/2$ (for odd M), two subclasses are formed. It is possible to continue

splitting each of these subclasses according to the same scheme receiving four subclasses. At the classification stage, the vector closest to one of the subclasses in the sense of the decisive rule (7.6) is considered to belong to an initial class.

If the initial number of available training vectors remains large even after separating into subspaces, the reduction of the $\mathbf{X}_k, \mathbf{X}_l$ matrices' dimension can be needed. Usually, this need relates to the limit of the computational resource. For class or subclass matrices dimension reduction, we can use a procedure similar to that described above. The difference is that instead of vector addition at each stage, we exclude vectors. The procedure is constructed as follows.

Let M be a number of training vectors in an initial set of kth class and M of conjugacy indexes are calculated for all vectors $\mathbf{x}_j, j = \overline{1, M}$ of kth class with subspace formed by all remaining vectors of the same class:

$$R_k(\mathbf{x}_j) = (\mathbf{x}_j^T \overline{\mathbf{X}}_k [\overline{\mathbf{X}}_k^T \overline{\mathbf{X}}_k]^{-1} \overline{\mathbf{X}}_k^T \mathbf{x}_j)(\mathbf{x}_j^T \mathbf{x}_j)^{-1}. \tag{7.32}$$

Here $\overline{\mathbf{X}}_k$ is $N \times (M-1)$-matrix formed by vectors of kth class without vector $\mathbf{x}_j$. We exclude from the kth class the vector $\mathbf{x}_j$, for which index $R_k(\mathbf{x}_j)$ reachs its maximum:

$$R_k(\mathbf{x}_J) = \max_{j=\overline{1,S}} R_k(\mathbf{x}_j). \tag{7.33}$$

The described procedure is applied for all classes, $k = \overline{1, K}$. Further, we exclude one vector from each class on each iteration in accordance with expressions (7.32) and (7.33). Hence, the dimensions of training subspaces for all classes are reduced per unit at each iteration. The iteration process continues until recognition quality stays acceptable for the required number of vectors.

The described technology is based on the following reasoning. If some number of vectors having the maximum conjugation is excluded, high informativity of a support subspace remains. The feature vectors, which are "similar" to the excluded one, are recognized confidently. At the same time, the vectors having low conjugacy indexes to their own class usually contain important information about the class. Their remoteness from the class in the sense of the conjugation index can be objectively related to the complexity of feature vector distribution in feature space. Such feature vectors usually

cause a linear inseparability in the Support Vector Machine (SVM) approach. In SVM linear separability can be restored by an enlargement of feature space by means of kernel functions. In our method, it is not necessary to use kernel functions. In our case, the recognition quality is closely related to these remote informative vectors of a support subspace.

7.6 Computing Aspects of Realization

To perform the class separation procedure and vector removal procedure mentioned above, it is necessary to carry out scanning of a large number of variants. For these procedures, conjugation indexes are to be calculated many times. Time spent for computations of the conjugation indexes depends on class size M. We can calculate the conjugation indexes by using (7.2) or (7.3). This possibility is based on equalities (7.17), which establishes the relationship between conjugation indexes (7.2) and (7.3). Therefore, it is important to establish the dependence of a calculation volume for indexes (7.2) and/or (7.3) on number M.

The decisive rules (7.6) and (7.7) based on conjugation indexes differ only by matrixes $\mathbf{Q}_{k,R}$ (7.4) and $\mathbf{Q}_{k,S}$ (7.5). Therefore, it is enough to compare the computing complexity of the formation of these matrixes [17]. For this, we count the number of the arithmetic operations for calculating each of these matrixes.

Total number of operations for the calculation of a matrix $\mathbf{Q}_{k,R}$ is

$$n = n_I + n_{II} + n_{III}, \tag{7.34}$$

where

$$\begin{aligned} n_I &= M^2 \cdot (N + (N-1)), \\ n_{II} &= \frac{M(M-1)^2}{2} + M^2 + 2M^3, \\ n_{III} &= N[M(M + (M-1)) + N(M + (M-1))]. \end{aligned}$$

Here n_I is the number of operations for calculation of the matrix $\mathbf{X^T X}$, n_{II} is number of operations for inversion this matrix by Holetsky's method, and n_{III} is number of operations at the final stage of calculation $\mathbf{X}[\mathbf{X^T X}]^{-1}\mathbf{X^T}$.

The number of operations for calculation of a matrix $\mathbf{Q}_{k,S}$ includes two terms. The first term is a number of the operations n_G for formation of a matrix $\mathbf{T}$ by a linear equations systems solution and a normalization of vectors. The second one is a number of the operations n_{TT} for calculation of the product of matrices $\mathbf{T}_0\mathbf{T}_0^T$:

$$n_S = n_G + n_{TT}, \tag{7.35}$$

where

$$n_G = \sum_{i=M}^{N-1} \left[\frac{i(i^2+3i-1)}{3} + M(N-(i+1)) \right] + 3N^3.$$

Figure 7.1 shows, as an example, the dependence of a number of the arithmetic operations on M (at fixed $N = 40$) for the procedure of conjugation indexes (7.4) and (7.5) calculation. This figure shows that it is more favorable to use the conjugation index (7.5) for large values of M.

The intersection point of curves on the plot corresponds to the boundary value of M. It is more favorable to use the conjugation index with zero space of a support subspace (7.5) for values greater than the boundary value.

During recognition, computational cost is caused by the calculation of $\mathbf{x}^T\mathbf{Q}\cdot\mathbf{x}$. Using this expression, it is easy to determine that the number of computing operations n (multiplication and addition) for

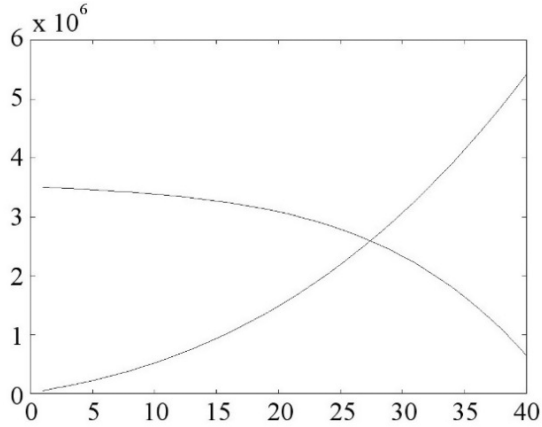


Figure 7.1. Dependence of arithmetic operations number from M at $N = 40$.

both calculations of conjugation indexes (7.4) and (7.5) is the same and equal

$$n = (2N - 1)N. \tag{7.36}$$

If dimensionality N of the feature vectors is large, computations are time-consuming. Reducing decision time often is one of the most important requirements in the systems for objects' recognition in images. Let us consider a calculation reduction during the recognition. In particular, it is possible to reduce the computational effort by representing matrix $\mathbf{Q}_{k,R}$ as (7.12) and/or $\mathbf{Q}_{k,S}$ as (7.5). In both cases, we must multiplicate the matrixes made of own eigenvectors corresponding to non-zero and zero eigenvalues, respectively.

We have to multiply matrixes $\mathbf{T}_\lambda$ and $\mathbf{T}_\lambda^T$ of size $N \times M$ and $M \times N$ for receiving $\mathbf{Q}_{k,R}$. It requires

$$(2M - 1)N^2 \tag{7.37}$$

operations of multiplication and addition. The matrix $\mathbf{Q}_{k,S}$ can be obtained by multiplication of matrixes $\mathbf{T}_{k,0}$ of size $N \times (N - M)$ and $\mathbf{T}_{k,0}^T$ of size $(N - M) \times N$ from (7.5). It requires

$$(2(N - M) - 1)N^2 \tag{7.38}$$

operations of multiplication and addition.

From (7.37) and (7.38), it can be established that the smaller number of operations using the conjugation index with zero space is possible only at $N < 2M$. If feature vector components are pixels of a digital image this condition usually is not satisfied. Therefore, we use a conjugation index (7.4) in the applications, which are considered in the following.

7.7 Examples of Face Recognition Algorithms

A huge number of works are devoted to the problem of face recognition. It is not possible to make even a short review of all of them within this chapter. Nevertheless, we note that in recent years the following approaches are the most popular: convolutional neural network [21], elastic graph matching [13], and active appearance models [3]. Perhaps, within the specified approaches sometimes it is

possible to apply a conjugation index as a proximity measure. However, so far authors have no examples of research in these directions. We investigated the application of the conjugation index as a proximity measure only for cases, when components of feature vectors are image pixels and when for the formation of a feature space the Principal component analysis (PCA) method is used.

PCA is one of the most known and popular methods. It is a statistical procedure, which uses an orthogonal transformation to convert a set of observations into a set of linearly uncorrelated variables, called principal components. In a problem of face recognition, it is applied to represent the face images by orthogonal vectors, which are called eigenface. Eigenfaces are defined once in the set of the training data. In the process of recognition, the face images are present by the weighed combination of these eigenfaces. If the number of eigenfaces is greater than a face image number, it is possible to apply the conjugation index as the proximity measure in the decisive rule. In the following, the experimental results are given for two cases: when components of the feature vectors are pixels of digital images and when the feature vectors are set in a space of the eigenfaces.

We present the experimental results of studies of the algorithms, constructed on the basis of conjugation indexes with the use of the standard ORL database [19, 20, 28]. This database was created in 1992–1994 in the research laboratory AT&T of Cambridge University (ATT). The database contains the face images of 40 people. There are 10 different observation views with arbitrary mimics for each person. Thus, the database contains 400 images. In Figure 7.2, as an example, face images of five persons from the ORL database are given.

The feature vectors were created by line-by-line scanning of face images. The size of images of the original ORL database is equal to 112×96. Thus, each image was presented by a vector with a

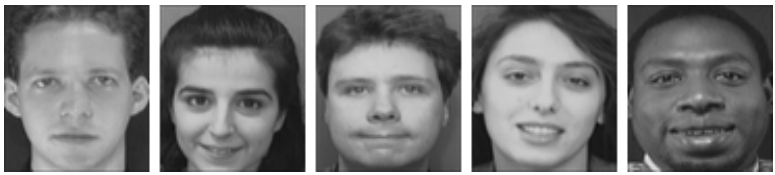


Figure 7.2. Examples of images from ORL base.

dimensionality of 10752×1. Matrixes $\mathbf{X}_k k = 1 \ldots 40$ were formed from vector sets for each class.

All database was separated into two parts: training and testing with 200 images in each, i.e. with five face images of each person. We used from one (40 images) to five (200) face images from the training set to establish the dependence of a recognition quality on a number of vectors in a class. We did not change the number of the test images for all experiments. In Figure 7.3, as an example, five face images are given for one person (class) on which 10752×5-matrix $\mathbf{X}_k$ was created.

In experiments, we used the classifier based on the use of the conjugation (7.4) with support subspace. For each class, we formed matrixes $\mathbf{X}_k$ and $\mathbf{Q}_{k,R}$ (support subspaces) from training vectors as described in Section 7.5. The decision about belonging some vector of the testing set to mth class was made using decision rules (7.6). The test dataset included 200 face images, i.e. five face images of each person. We need to emphasize that these images weren't used for the formation of matrixes $\mathbf{Q}_{k,R}$.

Figure 7.4 shows the recognition results for various proximity measures for the case when the feature vector components are image pixels. As one would expect, an increase in the number of training vectors in each class leads to an increase in the correct recognition percentage.

In Figure 7.5, similar results are presented for a case when the feature vectors are formed in space of eigenfaces (PCA) [12]. It should be noted that when in each class only one training vector, the conjugation index coincides with a cosine metric. In this case, the classification quality is identical. The conjugation index gives usually higher recognition quality with an increase in the vector number in each class. At the same time, the recognition quality can decrease with the inclusion of a person's image with very different foreshortening.



Figure 7.3. Example of images, belonging to one class.

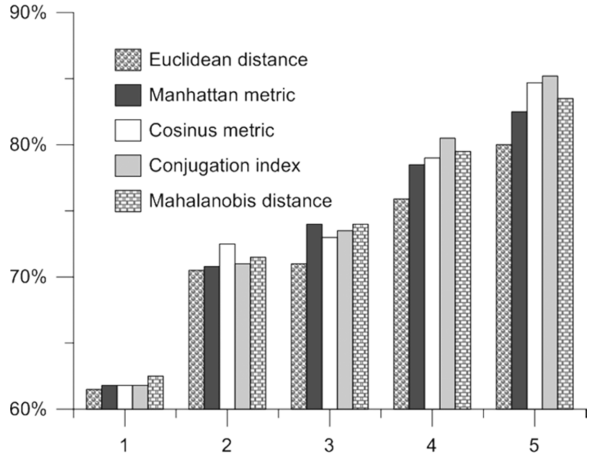


Figure 7.4. Dependence of recognition quality on number of the training vectors, when using as features the pixel brightness values.

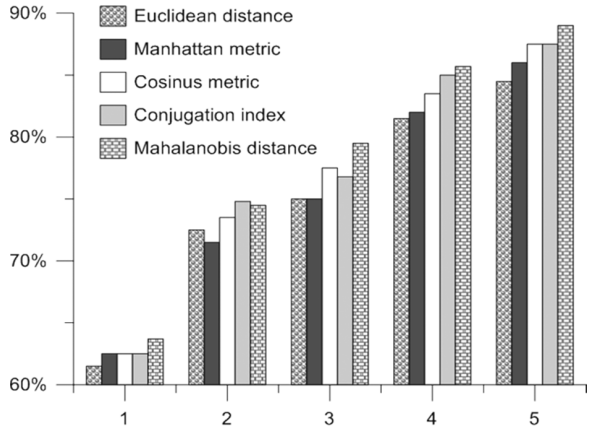


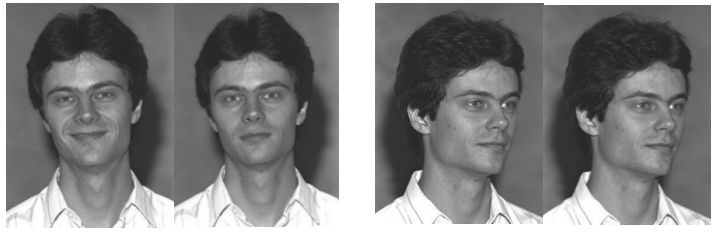
Figure 7.5. Dependence of recognition quality on number of the training vectors when feature vectors are formed in space of eigenfaces (PCA).

This problem can be solved by separating the classes into subclasses.

We made an experiment with the use of the FERET database [4] to check the efficiency of the classes' separation algorithm. We used the algorithm described in Section 7.6. In Figure 7.6, as an example, four of the used images are shown. The vectors corresponding to these images were separated into two subclasses. It can be seen, that

images received at different angles of observation are combined into different subclasses.

We compared the quality of the recognition with separation for cases of use of the conjugation index and Euclidean distance [17]. The training vector set is separated into two and three subclasses. The classification was carried out using support subspace according to the algorithm described above. Research results of recognition quality for splitting of the training classes for different numbers of subclasses are given in Figure 7.7. In particular, the dependence of the recognition percent on the number of clusters for cases of the use of the conjugation index and Euclidean distance is shown. Improvement of the recognition quality is gained for both the proximity measures, however it is more significant at the conjugation index.



(a) (b)

Figure 7.6. Illustration of images splitting into two classes.

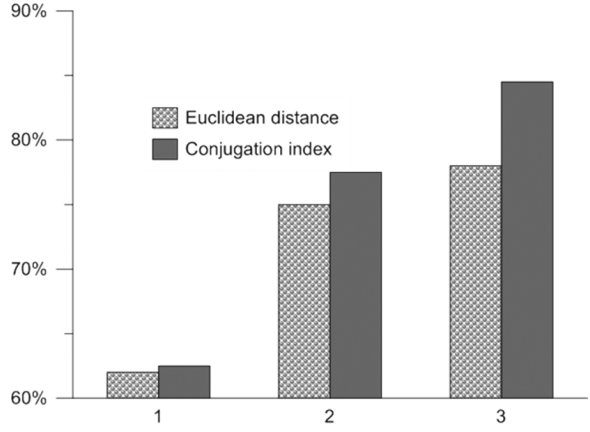


Figure 7.7. Dependence of recognition quality on number of subclasses in a class.

In conclusion, we emphasize that an important advantage of the conjugation index in comparison with the elastic graph matching and active appearance model methods is the possibility of recognition directly on the initial image without preliminary marking (graphs or points). These procedures are time-consuming and often carried out manually. The advantage of the conjugation index in comparison with neural networks is that addition of the one face image to the training set doesn't demand retraining. In our case, just one vector is added to the matrix.

7.8 Thematic Classification of Hyper-Spectral Images

The problem of thematic classification of hyperspectral images has attracted a lot of attention relatively recently. The multispectral and hyperspectral images received at remote sensing of a terrestrial surface allow solving a number of thematic classification problems for geology, climatology, forestry, oceanology, land use, control of water resources, etc. The number of publications on this topic is rapidly growing [18, 23, 33].

The problem of thematic classification is formulated as follows. Let

$$\mathbf{x}_j = [x_1(j), x_2(j), \ldots, x_i(j), \ldots, x_N(j)]^T \tag{7.39}$$

be $N \times 1$-vector, which presents jth "hyperpixel" of an image, where $x_i(j)$ is the reflection intensity value in ith spectral band and jth point of the hyperspectral image. The task is to construct a decision function $f : R^n \mapsto \{0, 1, 2, \ldots, k\}$ to identify which class each vector $\mathbf{x}_j$ belongs to.

It is assumed that M of training vectors for each class are given, i.e. $N \times M$matrix

$$\mathbf{X}_k = [\mathbf{x}_1(k), \mathbf{x}_2(k), \ldots, \mathbf{x}_M(k)], \quad k = \overline{1, K}, \tag{7.40}$$

is known. Thus, the problem of thematic classification of hyperspectral images is submitted by the standard formulation, which is described e in Section 7.3.

We used an image available in freeware MultiSpec to test the efficiency of the suggested method. The image had been obtained

within the project AVIRIS (Airborne Visible/Infrared Imaging Spectrometer). The image shows the test field "Indian Pines" located in the north-west of Indiana, the USA. The size of the image taken by AVIRIS is 145×145 pixels, where each pixel contains 200 spectral samples in the range of 0.4–2.5 mkm. The hyperspectral image is separated and marked of 16 classes. It should be noted that in this case (as and other cases), the feature vectors should have the same sizes, and their components should have the same spectral range. There is an unmarked area of the image, which does not refer to any of the 16 classes. We did not use this area in the experiment. Figure 7.8 shows the test image, where the unmarked area is white in color.

The most popular software, which has a great number of functions for analyzing hyperspectral images, is ENVI [26]. The method of spectral angle is one of the algorithms suggested in this software package [19,28]. In [32] we have compared a recognition method based on the use of the conjugation index to the method of a spectral corner. The accuracy of recognition by support subspaces method has been reached more than 13% above.

Results of recognition accuracy comparison by conjugation index method with support vector machine (SVM) method are given in this chapter. We have applied the technology for the formation of support subspaces described above. The SVM method was carried out by program version 2017 (a) which is available in the open package of Matlab. The choice of SVM for comparison was dictated by the fact that this method is used widely in problems of thematic classification [19, 28].

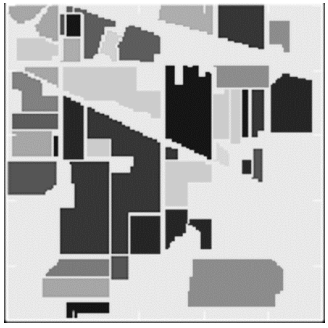


Figure 7.8. The test image "Indian Pines".

In the training stage, we have included to the technology the following algorithms of preliminary processing to reduce computing complexity and to improve the quality of classification:

(1) separation of the classes into subclasses,
(2) subtraction of an average vector,
(3) removal of almost-linearly-dependent vectors,
(4) the procedure of components weighing.

The procedure of separation of the classes into subclasses (point 1) was carried out as described in Section 7.6. We split each class into four subclasses, but we did not separate classes, which contain less than 50 training vectors.

The average vector (point 2) was obtained by calculating the average value for each layer (spectral band) of the image. Subtraction of an average vector from a hyperspectral image increases angles between the vectors belonging to different classes. This leads to better separability of the classes.

Removal of almost-linearly-dependent vectors (point 3) is carried out to avoid ill-conditioning of the matrix $\mathbf{X}_k$ in (7.4). Removal of such vectors doesn't lead to loss of information. The procedure is constructed by consecutive removal of one vector from a pair of the most neighboring vectors. To do so, on each step among columns of matrix $\mathbf{X}_k$ we defined a couple of vectors $\mathbf{x}_i, \mathbf{x}_j$ for which

$$R^*_{i,j} = \max_{\forall i,j=\overline{1,M}} R_{i,j}, \tag{7.41}$$

where

$$R_{i,j} = \langle \mathbf{x}_i^T \mathbf{x}_j \rangle / \|\mathbf{x}_i\| \|\mathbf{x}_j\|, \quad i,j = \overline{1,M}. \tag{7.42}$$

Further, one of these vectors is removed from the support subspace. The procedure repeats for vector set, which has remained after removal. The process stops when $R^*_{i,j}$ is equal to the threshold value R_{tresh} or when the predetermined number of vectors M is reached.

The task of components weighing (point 4) can be formulated as an optimizing problem: to determine positive definite $N \times N$-matrix $\mathbf{G}$ for the so-called *weighed conjugation index* determined as

$$R_k(\mathbf{G}, j) = \mathbf{x}_j^T \mathbf{Q}_k(\mathbf{G}) \mathbf{x}_j (\mathbf{x}_j^T \mathbf{x}_j)^{-1}, \tag{7.43}$$

where $\mathbf{Q}_k(\mathbf{G}) = \mathbf{G}\mathbf{X}_k[\mathbf{X}_k^T\mathbf{G}^2\mathbf{X}_k]^{-1}\mathbf{X}_k^T\mathbf{G}$, $k = \overline{1,K}$, and $\mathbf{G} = \mathrm{diag}(g_1,\ldots,g_i,\ldots,g_N)$:

$$\sum_{i=1}^{N} g_i = N. \tag{7.44}$$

The results of the thematic classification of the test image "Indian Pines" using the main stages of the described algorithm can be found in our work [14].

7.9 Technologies of the Synthetic Aperture Radar Images Recognition

In this section, we consider the application of the conjugation index method to the recognition of synthetic aperture radar (SAR) images. The SAR images frequently are used in a problem of ground target recognition. There is growing interest in using SAR images for object recognition which can be explained by the low sensitivity of observed data to cloud cover that restricts the application of visible and infrared data [8,14]. For the illustration of our results, we use the popular ground target images dataset: Moving and Stationary Target Acquisition and Recognition (MSTAR) [25] that was obtained during experiments by DARPA. The SAR images of ground targets were formed by an airborne radar with a resolution of 0.3 m at the different poses. The discussion and results of MSTAR image recognition using different principles in [22, 30] were presented.

The sensitivity of the SAR images to different target poses [6, 15] is the most important factor, which has influence at the recognition stage. There are two popular methods of image feature extraction: finding target pose angles [29] and finding object scattered centers in the images [2] usually are considered. Principal component analysis (PCA) is frequently used for reducing the vector feature space [1, 29]. For example, in [1], it was applied to reduce a feature space of 6400 to 150.

Various SAR image recognition methods are widely used, such as support vector machine (SVM), neural network, template matching, MINACE, C5.0 decision tree classifier, QUEST classifier, and so on. SVM is the most popular recognition method. For example, in [27],

a comparative analysis was carried out for four methods: SVM, normal Bayes (NB), classification and regression tree (CART), and K-nearest neighbor (KNN). In [8], the classification accuracy was compared for the SAR images and optical data using the SVM technique. The authors indicate the main disadvantage of SVM which is that it is complicated to optimize the algorithm parameters.

There are many publications [1, 6, 15, 24] with the analysis of different approaches to object recognition using the MSTAR dataset. We chose these objects in order to compare our results with the results of other works [1, 29]. Analysis of well-known works shows that the main problem of SAR image recognition is a high sensitivity to the object pose changing. In this section, we continue to examine the conjugation index method and take into account specified problems of SAR image recognition.

Here, as well as earlier, we use pixel-based features of the SAR images. The SAR image with index j we present as a feature vector, with the size of $N \times 1$:

$$\mathbf{x}_j = [x_1, x_2, \ldots, x_i, \ldots, x_N], \tag{7.45}$$

where components are $N = W \times H$ pixels and W and H are image sizes. Vectors $\mathbf{x}_j$ through lines or columns of images are formed by scanning. It is assumed that M vectors were formed, which can be used for training. Since the observation parameters have errors, it is necessary to use a sufficiently large number of feature vectors for training, which form $N \times M$-matrixes of classes as

$$\mathbf{X}_k = [\mathbf{x}_1(k), \mathbf{x}_2(k), \ldots, \mathbf{x}_j(k), \ldots, \mathbf{x}_M(k)], \quad k = \overline{1, K}. \tag{7.46}$$

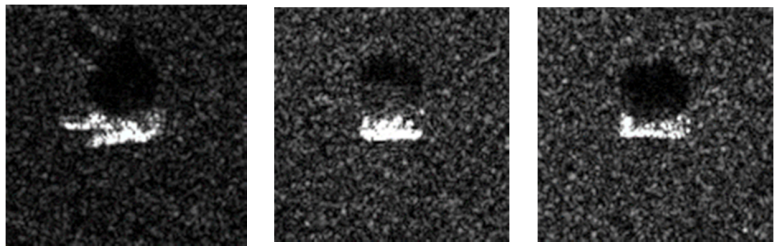
The vectors (7.45) and matrixes (7.46) are used within statements (7.2)–(7.7) described in Section 7.3.

Thus, the problem of recognition of synthetic aperture radar (SAR) images leads to the statement described in Section 7.3. Image preparation procedures were implemented using SAR images of the MSTAR dataset. We carried out additional pre-processing of the SAR images as described in [7]. In Figures 7.9(a)–7.9(c), the SAR image examples are presented of three objects, which were exported from the MSTAR dataset: TANK, BMP, and BTR. Further, we investigate the recognition quality of the conjugation index method for three specified objects. We concentrate attention on two major

aspects: research of the recognition quality sensitivity to the object pose changing and construction of support subspaces that provide high recognition quality.

The dataset for TANK and BMP contains images for three different models. We purposely carried out experiments under the same models and conditions as are used in [2] to compare results. In our experiments, training support subspaces are formed for all $K(K-1)/2$ pairs of classes, where $K = 3$ is a common number of classes in this task. The algorithm of multiclass recognition is executed in a manner of one-to-one classification. We studied the sensitivity of object pose changing in the first experiment. In this experiment, we used full a training set, without the separation of classes into subclasses. Table 7.1 shows the obtained results.

Also, we carried out the separation of classes into two and four subclasses. Tables 7.2 and 7.3 show detailed information on the results of object recognition obtained with the separation of subclasses. We obtained a better result than shown in [5] (96.1%), although about 10% of inappropriate feature vectors from the test set



(a) (b) (c)

Figure 7.9. Using SAR images of (a) tank; (b) bmp; (c) btr.

Table 7.1. Confusion matrix without clustering.

		Estimated class			
		bmp	btr	Tank	Accuracy
Actual class	bmp	538	6	43	91.6%
	btr	0	196	0	100%
	Tank	2	1	579	99.4%
General result on all classes					96.1%

Table 7.2. Confusion matrix with clustering on two subclasses.

		Estimated class			
		bmp	btr	Tank	Accuracy
Actual class	bmp	556	9	22	94.7%
	btr	0	196	0	100%
	Tank	2	1	579	99.4%
General result on all classes					97.5%

Table 7.3. Confusion matrix with clustering on four subclasses.

		Estimated class			
		bmp	btr	Tank	Accuracy
Actual class	bmp	555	9	23	94.5%
	btr	1	195	0	99.4%
	Tank	0	1	581	99.8%
General result on all classes					97.5%

by the authors of work [2] were removed. Our method works better without removing inappropriate vectors.

In comparison, we have also carried out recognition using the SVM method with the same experimental conditions. Results are given in Table 7.3. Though the result of the SVM method improves when classes are separated into subclasses, nevertheless it doesn't achieve the result obtained by our method. It means that the recognition quality by SVM is more sensitive to object pose changing.

The second series of experiments was carried out to answer the question: is it possible to use small training datasets without significant loss of recognition quality? Figure 7.10 shows the recognition results obtained with support subspaces of different sizes.

Here, the recognition result with SVM was given for comparison (dotted curve 4). Note that SVM results were found nearest to the published results in [2]. Curves 1 (dotted) and 2 (solid) show the recognition quality using support subspaces of different dimensions. Curve 1 was obtained by forming support subspaces of the initial training vector set. Curve 2 is received for a case of preliminary removal from each class four vectors according to expressions (7.32)

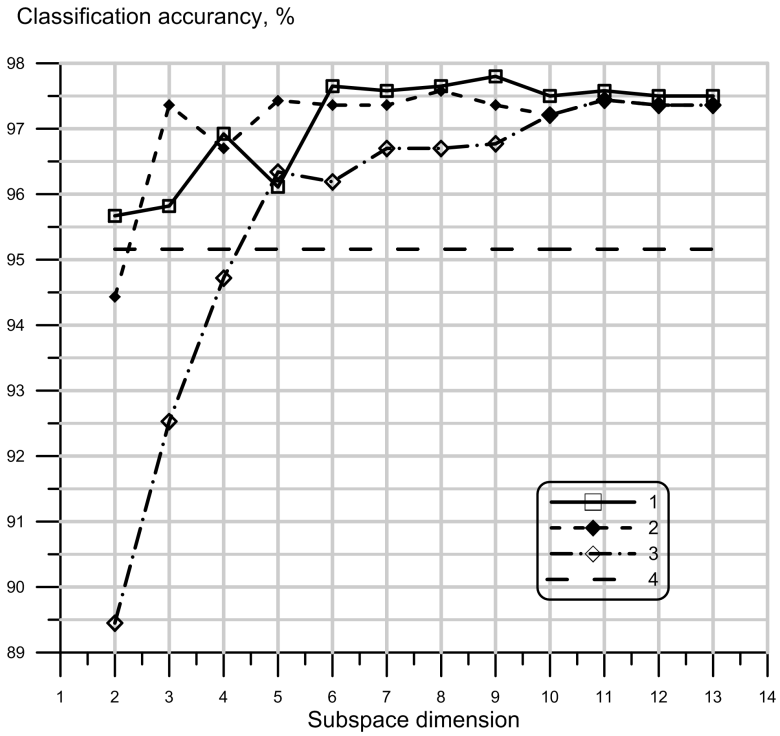


Figure 7.10. Graphs of recognition quality depending on support subspace dimension.

and (7.33). After comparing curves 1 and 2, we see that the accuracy of recognition more often than not is higher in a case when from a support subspace the vectors having the maximum of the conjugation index to their own class are removed. This result explains the effect of low sensitivity when we use the conjugation index method in recognition of object pose changing (see Table 7.4).

As expected, using support subspaces of small dimensions leads to a decrease in recognition quality when the number of subclasses decreases. In particular, the percentage of correct recognition using subspaces with dimension $d = 2$ (support planes) changes from 81.3% without clustering to 95.6% with clustering on 16 subclasses that is higher than using SVM. We reached 97.8% of correct recognitions using the subspaces support method when the subspace dimension was equal to nine (curve 1, Figure 7.10) and a training set was divided into 16 subclasses.

Table 7.4. Recognition results using conjugation index method and SVM.

	Number of subclasses in a class		
	0	2	4
CIM (%)	96.1	97.5	97.5
SVM (%)	92.8	94.7	95.1

Comparison with other results obtained on the same dataset with recognition and training on the same models produces interesting results. We compared our results to the results of four other methods: SVM shows a recognition result of 94.8%, neural network of 94.0%, template matching methods of 89.7% and that described in [29]. According to [2], the percent of correct recognition is 96.1%, obtained using AdaBoost with radial bases function method. Deep learning on the MSTAR dataset obtained the result of 96.6% correct recognitions [24] that is less than our result.

7.10 Conclusion

The conjugation index method is preferable to apply when the use of other feature systems (geometrical, etc.) is difficult because of the image features and/or a feature component number is big in comparison with number of the training vectors.

As shown above, the conjugation index is the sum of the cosine squares of the "angles" between the recognizable vector and eigenvectors corresponding to non-zero eigenvalues of the matrix composed from the vectors of a support subspace. Therefore, the recognition by conjugation index method provides high-quality recognition because of the use of only the direction of vectors but not their length. This method can be the worst in a case when the components of vector variations for different classes are similar in the thought that they differ in intensity. It explains the fact that on the test "Indian Pines" the percent of recognition is lower than that of the recognition of the radar images.

It must be emphasized that separability of the classes by the conjugation index is possible even, if the separability in traditional sense is absent.

Acknowledgments

The publication is realized in the framework of the Ministry of Education and Science of Russian Federation (project # 2.7891).

References

[1] Amoon, M. and Rezai-Rad, G.A. (2013) Automatic target recognition of synthetic aperture radar (SAR) images based on optimal selection of Zernike moments features. *IET Computer Vision*, 8(2), 77–85.

[2] Carvalho, O.A. and Meneses, P.R. (2000) Spectral correlation mapper: An improvement on the spectral angle mapper (SAM). *Summaries of the 9th JPL Airborne Earth Science Workshop*, Vol. 9 (JPL Publication), pp. 00–18.

[3] Chaban, L.N., Vecheruk, G.V., Kondranin, T.V., Kudriavtsev, S.V., and Nikolenko, A.A. (2012) Modeling and thematic processing of images identical to the imagery from workable and preparing for the space launch hiperspectral remote sensors. *Current Problems in Remote Sensing of the Earth from Space*, 9(2), 111–121 (in Russian).

[4] Deng, L., Yan, Y.N., and Sun, C. (2015) Use of sub-aperture decomposition for supervised PolSAR classification in urban area. *Remote Sensing*, 7(2), 1380–1396.

[5] Diemunsch, J. and Wissinger, J. (1998) MSTAR model-based automatic target recognition: search technology for a robust ATR. *Proceedings of SPIE. Algorithms for Synthetic Aperture Radar Imagery V*, Vol. 3370, pp. 481–492.

[6] Dusseux, P., Corpetti, T., Hubert-Moy, L., and Corgne, S. (2014) Combined use of multi-temporal optical and radar satellite images for grassland monitoring. *Remote Sensing*, 6(7), 6163–6182.

[7] Edwards, G.J., Cootes, T.E., and Taylor, C.J. Face recognition using active appearance models. *Proceedings of European Conference on Computer Vision*, pp. 581–595.

[8] Research System Inc. (2004) *ENVI 4.1 User's Guide*, 1150 p.

[9] Feng, J., Cao, Z., and Pi, Y. (2014) Polarimetric contextual classification of PolSAR images using sparse representation and superpixels. *Remote Sensing*, 6(8), 7158–7718.

[10] FERET Face Database. http://www.itl.nist.gov/iad/humanid/colorferet/.

[11] Fursov, V. (2000) Training in pattern recognition from a small number of observations. *Proceedings of 15th International Conference on Pattern Recognition, ISPR*, Vol. 2, pp. 119–121.

[12] Fursov, V. and Kozin, N. (2007) Recognition through constructing the eigenface classifiers using conjugation indices. *Proceedings of Advanced Video and Signal Based Surveillance*, pp. 465–469.

[13] Fursov, V.A., Kulagina I.A., and Kozin, N.E. (2007) Building of classifier based on conjugation indexes. *Proceedings of 5th International Conference on Machine Learning and Data Mining*, pp. 231–235.

[14] Fursov, V., Bibikov, S., Zherdev, D., and Kazanskiy, N. (2020) Thematic classification with support subspaces in hyperspectral images. *International Journal of Engineering Systems Modelling and Simulation*, 11(4), 186–193.

[15] Fursov, V., Minaev, E., Zherdev, D., and Kazanskiy, N. (2017) Support subspaces method for recognition of the synthetic aperture radar images using fractal compression. *International Journal of Advanced Robotic Systems*, 14(5), 1–8.

[16] Gao, G., Zhao, L., Zhang, J., Zhou, D., and Huang, J. (2008) A segmentation algorithm for SAR images based on the anisotropic heat diffusion equation. *Pattern Recognition*, 41(10), 3035–3043.

[17] Keydel, E.R., Lee, S.W., and and Moore, J.T. (1996) MSTAR extended operating conditions a tutorial. *Proceedings of SPIE. Algorithms for Synthetic Aperture Radar Imagery III*, Vol. 2757, pp. 228–242.

[18] Kottke, D.P., Fiore, P.D., Brown, K.L., and Fwu, J.K. (1998) A design for HMM-based SAR ATR. *Proceedings of SPIE. Algorithms for Synthetic Aperture Radar Imagery V*, Vol. 3370, pp. 541–551.

[19] Kozin, N. and Fursov, V. (2007) Building of classifiers based on conjugation indices. *Proceedings of International Conference on Machine Learning and Data Mining*, pp. 136–143.

[20] Kuznetsov, A.V. and Myasnikov, V.V. (2014) A comparison of algorithms for supervised classification using hyperspectral data. *Journal of Computer Optics*, 38(3), 494–502 (in Russian).

[21] Lawrence, S., Lee Giles, C., Tsoi, A.C., and Back, A.D. (1997) Face recognition: A convolutional neural-network approach. *IEEE Transactions on Neural Networks*, 8(1), 98–113.

[22] Mossing, J.C. and Ross, T.D. (1998) An evaluation of SAR ATR algorithm performance sensitivity to MSTAR extended operating conditions. *Proceedings of SPIE. Algorithms for Synthetic Aperture Radar Imagery V*, Vol. 3370, pp. 554–565.

[23] Moving and Stationary Target Acquisition and Recognition (MSTAR). https://www.sdms.afrl.af.mil/index.php?collection=mstar.

[24] Qian, Y., Zhou, W., Yan, J., Li, W., and Han, L. (2014) Comparing machine learning classifiers for object-based land cover classification using very high resolution imagery. *Remote Sensing*, 7(1), 153–168.

[25] Schowengerdt, R.A. (2006) *Remote Sensing: Models and Methods for Image Processing* (Academic Press, London, UK), 560 p.
[26] Shafri, H.Z.M., Affendi, S., and Shattri M. (2007) The performance of maximum likelihood, spectral angle mapper, neural network and decision tree classifiers in hyperspectral image analysis. *Journal of Computer Science*, 3(6), 419–423.
[27] Sun, Y., Liu, Z., Todorovic, S., and Li, J. (2005) Adaptive boosting for synthetic aperture radar automatic target recognition. *IEEE Transactions on Aerospace and Electronic Systems*, 43(1), 112–125.
[28] Tansini, L. and Viera, O. (2005) New measures of proximity for the assignment algorithms in the MDVRPTW. *Journal of the Operational Research Society*, 57(3), 241–249.
[29] Uzdin, D.Z. (2016) *New Measures of Proximity, Functions of States and Decision Rules in the Theory of State Recognition (Statistic Classification),* 2nd Ed. updated and revised (MAXS Press), 128 p.
[30] Wiskott, L., Fellous, J.M., Krüger, N., and Von Der Malsburg, C. (1997) Face recognition by elastic bunch graph matching. *Intelligent Biometric Techniques in Fingerprint and Face Recognition. Proceedings of International Conference on Computer Analysis of Images and Patterns*, pp. 456–463.
[31] Zhang, P., Li, M., Wu, Y., Gan, L., Liu, M., Wang, F., and Liu, G. (2012) Unsupervised multi-class segmentation of SAR images using fuzzy triplet Markov fields model. *Pattern Recognition*, 45(11), 4018–4033.
[32] Zhao, Q., Principe, J.C., Brennan, V., Xu, D., and Wang, Z. (2000) Synthetic aperture radar automatic target recognition with three strategies of learning and representation. *Optical Engineering*, 39(5), 1230–1244.
[33] Zherdev, D.A. and Fursov, V.A. (2014) Pattern recognition of electromagnetic field scattering from anthropogenic objects on underlying surface. *Proceedings of SPIE. Optics and Photonics for Information Processing VIII*, Vol. 9216, p. 92160Z.

© 2025 World Scientific Publishing Company
https://doi.org/10.1142/9789811267215_0008

Chapter 8

Target Shape Identification on Low-Count Images Using Expectation–Maximization Algorithms

Vyacheslav E. Antsiperov* **and Aleksandr S. Bugaev†**

Kotelnikov Institute of Radioengineering and Electronics of Russian Academy of Sciences, Moscow, the Russian Federation

**antciperov@cplire.ru*

†*bugaev@cplire.ru*

This work is devoted to the detection of target objects of a given shape on images of a special class. This class of images corresponds to the case of weak radiation registration and, accordingly, is characterized by a relatively low-count rate (images with a low-count rate). Such images are typical in deep space astronomy (optical, infrared radiation), computer and positron emission tomography (X-ray radiation), and thermography (infrared radiation). The low-count character of images leads to a small signal-to-noise ratio, low contrast, and fuzzy shapes of the detected objects. Moreover, in the case of a limited exposure time, registration noise, which inevitably appears in such images, demonstrates a pronounced Poisson character — it turns out to be both non-additive and significantly dependent on the image intensity. Thus, the well-known recognition methods developed for traditional images would unlikely work properly on images with a low-count rate. To overcome this problem, a new approach to object recognition in the class of images with a low-count rate has been developed. This chapter presents a new method

for identifying the shape of objects, synthesized within the framework of the proposed approach. In accordance with the principles of machine learning, the problem under consideration is solved as a statistical inference from the earlier observed training data and not as a traditional problem of parameter estimation.

Keywords: Low-count images, low contrast and fuzzy shapes, machine learning, weak radiation registration, photon-counting sensors, Poisson point process (PPP), maximum likelihood (ML) estimation, EM-like algorithms.

8.1 Introduction

This chapter is devoted to the detection of target objects of a given (based on precedents) shape on images of a special class. This class of images is generally associated with weak radiation registration and, therefore, characterized by a relatively low rate of counts (low-count images) [4]. Such images are common in astronomy (optical radiation), computer and positron emission tomography (X-ray radiation), fluorescence microscopy (ultraviolet), and thermography (infrared radiation).

Terahertz (THz) images, which have become very popular in the last few decades, are typical examples of low-count imaging. It is largely because of the lack of natural sources of THz radiation due to the so-called "terahertz gap" between microwave and infrared ranges and problems in developing sufficiently powerful artificial sources. As a result, THz imaging procedures are often performed slowly, pixel by pixel, resulting in a high price for one pixel and a small total number of them compared to conventional (video) images. Figure 8.1 shows an example of images taken in parallel by THz (passive illumination) and traditional cameras.

The low-count nature of the images results in a low signal-to-noise ratio, low contrast, and fuzzy shape of the detected objects. In combination with not too long exposure time, the registration noise inevitably generated in such images demonstrates a pronounced speckle-like Poisson character — it turns out to be both non-additive and significantly dependent on the image intensity. Thus, the known recognition methods developed for traditional images do not perform well on images with a low-count rate.

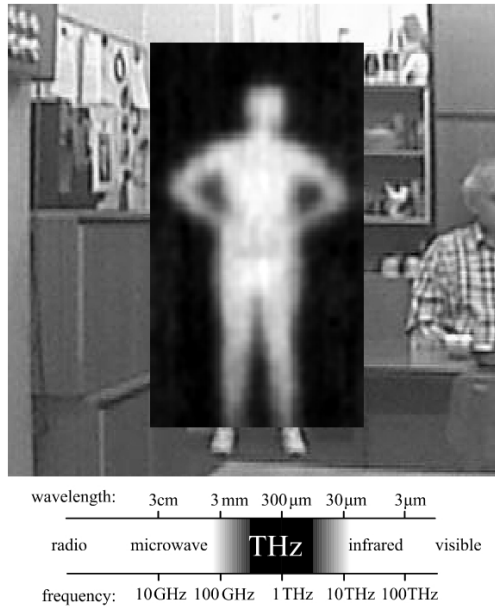


Figure 8.1. An image in the terahertz (THz) range (central black box) compared to a traditional image in the visible range (peripheral surroundings).

In view of the above circumstances, we have developed a new image analysis approach focused on the class of low-count images. Within the framework of this approach, we have proposed a new method for shape identification of the objects present in images. The proposed method is essentially focused on the machine learning paradigm. In accordance with the principles of machine learning, the problem considered is posed as the statistical inference via already observed data rather than the traditional statistics problem of parameter estimation. In other words, it is assumed that a finite set of training objects (precedents) has been registered, and for every such precedent, some description is assumed in terms of its shape. Thus, the problem is to recognize to which of them the analyzed object should be attributed.

However, before discussing the method itself, it is necessary to consider in more detail some aspects of the weak radiation registration. The goal of this consideration is to introduce the most adequate statistical formal description of the data registered and, ultimately, to reveal the specifics of the class of low-count images.

8.2 The Physical Basis of Weak Radiation Registration

In recent decades, there has been a growing interest in developing the so-called photon-counting image sensors [5]. This trend began already in the late 1960s with the development of CCD cameras and finally took hold in the early 1990s with the invention of solid-state CMOS image sensors. The increased interest in these technologies is due to the possibility of reducing the pixel pitch of video matrices, increasing frame rate, and the possibility of registering weak and extremely weak radiation. Due to these features, the registration process acquires a more pronounced quantum character, and in the limiting case, it becomes the detection of individual photons (photoelectrons). For this reason, the corresponding sensors are also called single-photon imaging devices, photon-counting array detectors, quanta image sensors (QIS), etc.

8.2.1 *Quantum semi-classical weak radiation registration*

The quantum character of registration, i.e. the possibility of detecting each released photoelectron and its immediate transferring into a microprocessor, substantially changes the basic concepts used in the theory and practice of image processing. For example, in classical applications of computer vision, the processing of incoming information is carried out in several steps. In the first step, a stream of photocounts is accumulated, then, based on them, the image is restored, and in the last step, the image is transferred to specialized procedures for extracting certain features, parameters of the objects, etc. On the contrary, when computer vision is realized with the help of photon-counting sensors, it is possible to synchronize the data recording process and the process of feature extracting directly from the stream of photocounts. So, in the case of photon-counting sensors, the need for an intermediate image reconstruction disappears and a new concept of "vision without an image" comes into being [3].

In this section, we formalize the (statistical) description of registered by photon-counting sensor data as the stream of photocounts (or, for short, counts). Using a discrete set of count coordinates

$\{\vec{x}_1, \vec{x}_2, \ldots, \vec{x}_n\}$, rather than a continuous image reconstruction, we formalize the problem in question as the problem of the best fitting of this set, considered as the realization of a certain point process, to the previously registered analogous point processes, which we call precedents.

8.2.2 *Poisson point process model of data registered*

To formalize the statistical description of registered radiation data as the stream of counts, it is necessary to choose such a mathematical model that most adequately relates the set of registered counts $\{\vec{x}_1, \vec{x}_2, \ldots, \vec{x}_n\}$ to the intensity $I(\vec{x})$ of the radiation incident on the sensitive surface Ω of the sensor. Here, $\vec{x} \in \Omega$ denotes continuous coordinates in some coordinate system on the Ω plane, and $\vec{x}_i$ are the coordinates of the pixel where the ith count is registered.

In the ideal case, when the size of the pixels can be considered arbitrarily small, in the so-called continuous model of the photodetection [6], the coordinates of the counts $\vec{x}_i$ are a set of random vectors (of random number n), even for a given non-random intensity $I(\vec{x})$. The most general mathematical model for such a phenomenon is a random point process [12]. From the semiclassical theory of photodetection, it is well known that among such processes the Poisson point process (PPP) simulates the photocounts generation in the most adequate manner [10]. A remarkable feature of PPP is that a complete statistical description of counts can be determined by the only function — the count intensity $\lambda(\vec{x})$ — all finite-dimensional probability distributions can be expressed in terms of this function [12]:

$$\rho(n, \vec{x}_1, \ldots, \vec{x}_n) = \frac{1}{n!} \prod_{i=1}^{n} \lambda(\vec{x}_i) \times \exp\left\{ - \int_{\Omega} \lambda(\vec{x}) d\vec{x} \right\}. \qquad (8.1)$$

Quantum theory specifies that the density $\lambda(\vec{x})$ in (8.1) is related by a simple expression to the intensity $I(\vec{x})$ of the incident radiation, namely, $\lambda(\vec{x}) = \eta T I(\vec{x})/h\bar{\nu}$, where η is the coefficient of quantum efficiency of the detector material, T is a frame readout time, $\bar{\nu}$ is the characteristic frequency of the incident radiation, and h is the Planck constant [6].

The relation, expressing $\lambda(\vec{x})$ through $I(\vec{x})$, together with the distributions (8.1), defines the model relating the input $I(\vec{x})$ of an ideal (continuous) detector to the statistics of its output — PPP counts $\{\vec{x}_i\}$. For real sensors with pixels of finite dimensions, the corresponding relation is obtained by integrating (8.1) over pixel surfaces, considering the exact number of photocounts $\{n_i\}$ registered by each of them. Note that the model of the real sensor will be close to the ideal model (8.1) in those cases when the probabilities of two or more counts ($n_i > 1$) are small compared with the probability of one photocount ($n_i = 1$). It is easy to show that the necessary condition for that is $\lambda(\vec{x})\sigma_i < 1$, where σ_i is the area of an individual pixel. Thus, when registering weak radiation $I(\vec{x}) \to 0$, or in high frame rate imaging $T \to 0$, or in the case of pixel $\sigma_i \to 0$, statistical description (8.1) is an adequate model for the photon-counting sensors.

For this reason, the set of finite-dimensional distributions (8.1) could be chosen as the basis for the statistical synthesis of recognition/identification methods for photon-counting sensors. However, one more step can be made in this direction if we slightly extend the formulation of the problem, namely, let us take as the identity relation of objects the similarity of the form of their radiation intensities regardless of the total brightness of each of them (see the discussion of this formulation in [1]). From the statistical point of view, this means that instead of the joint distributions (8.1), the conditional distributions of the count coordinates $\{\vec{x}_i\}$ for a given total number n should be chosen as the basis:

$$\begin{aligned} \rho(\vec{x}_1, \ldots, \vec{x}_n | n) &= \prod_{i=1}^{n} \rho(\vec{x}_i), \\ \rho(\vec{x}) &= \frac{\lambda(\vec{x})}{\int_\Omega \lambda(\vec{x}) d\vec{x}} = \frac{I(\vec{x})}{\int_\Omega I(\vec{x}) d\vec{x}}. \end{aligned} \tag{8.2}$$

The distributions (8.2) express the well-known property of the PPP that the conditional joint distribution of the counts decomposes into a product of identical distributions of each of them [12]. In other words, for a given n, the counts $\{\vec{x}_i\}$ are a set of independent identically distributed (iid) random vectors. Such samples are the source data for most popular statistical approaches, such as naïve Bayesian, which makes the given formalization of the problem very attractive. Moreover, the count distribution $\rho(\vec{x})$ (8.2) coincides directly with the (normalized) intensity $I(\vec{x})$ and does not depend on either the

detector material η or the carrier frequency $\bar{\nu}$ of the radiation or the registration time T and pixel area σ_i. In view of these remarkable properties of the distributions (8.2), they have a universal character, and for this reason, they were chosen as the statistical basis of our approach.

8.3 The Mathematical Basis of Statistical Learning

As noted in the introduction, the proposed approach to objects/target identification is focused on machine learning methods [8]. Therefore, the available training data (precedents) are also initially represented by registered (in the training step) sets of counts, and their statistics is also given by (8.2). However, the use of the precedent's direct registered data in identification procedure is not rational, both because of the low efficiency of resulting procedures and the wasteful use of computing resources (large amounts of data, respectively, large search times for target precedents, etc.). An obvious way here is to form, on the basis of the recorded data, some compact statistical characteristics of the intensity $I(\vec{x})$ of the precedent's radiation and then to use these characteristics to identify object set of counts. In order to substantiate this approach, we use some ideas from the field of statistical inference.

8.3.1 *Some definitions from statistical inference theory*

Let some objects be described by some complete set of (random) data $\vec{Y} = \{y_1, \ldots, y_{p+q}\}$, of which only a part $\vec{X} = \{x_1, \ldots, x_p\}$ is available (registered, observed data). Denote by $\vec{Z} = \{z_1, \ldots, z_q\}$ the remaining (hidden, latent) data that complement $\vec{X}$ to $\vec{Y}$ so that $\vec{Y} \sim \{\vec{X}, \vec{Z}$ [2]. Let the objects in question be labeled with labels from some finite set $\{M\}$. Then, a complete statistical description of the available data would be given by a finite set of conditional probability distributions with the densities $\rho(\vec{Y}|M)$ — the probabilities of some object's complete data $\vec{Y}$, provided that the object is labeled by M. The completeness of this description implies the fact that with the help of the distribution $\rho(\vec{Y}|M) = \rho(\vec{X}, \vec{Z}|M)$ all other probabilities can be found — marginal distributions of

observed/hidden data $\rho(\vec{X}|M)/\rho(\vec{Z}|M)$, their conditional distributions $\rho(\vec{X}|\vec{Z};M)/\rho(\vec{Z}|\vec{X};M)$, etc.:

$$\begin{aligned} \rho(\vec{X}|M) &= \int \rho(\vec{X},\vec{Z}|M)d\vec{Z}, \\ \rho(\vec{Z}|M) &= \int \rho(\vec{X},\vec{Z}|M)d\vec{X}, \end{aligned} \tag{8.3}$$

$$\begin{aligned} \rho(\vec{X}|\vec{Z};M) &= \rho(\vec{X},\vec{Z}|M)/\rho(\vec{Z}|M), \\ \rho(\vec{Z}|\vec{X};M) &= \rho(\vec{X},\vec{Z}|M)/\rho(\vec{X}|M). \end{aligned} \tag{8.4}$$

For the synthesis of meaningful procedures for identifying an object, i.e. identifying a label M from the observed data $\vec{X}$, it is useful to restrict the overall complete description $\rho(\vec{Y}|M)$ to some parametric model, namely, assuming that the complete data $\vec{Y}$ for all M are modeled by some universal parametric distribution with density $\rho(\vec{Y}|\vec{\theta})$, $\vec{\theta}=\{\theta_1,\ldots,\theta_r\}$, and the relationship of $\vec{Y}$ with a certain label M is given by *a priori* parameter distribution $\rho(\vec{\theta}|M)$, one can take the following Markovian refinement of the complete statistical description [2]:

$$\rho(\vec{Y}|M) = \int \rho(\vec{Y}|\vec{\theta})\rho(\vec{\theta}|M)d\vec{\theta}. \tag{8.5}$$

Note that in the case of singular *a priori* densities $\rho(\vec{\theta}|M) = \delta(\vec{\theta}-\vec{\theta}_M)$, where $\delta(\cdot)$ is the Dirac's delta function, the description $\rho(\vec{Y}|M)$ (8.5) reduces, as above, to the finite set of distributions $\{\rho(\vec{Y}|\vec{\theta}_M)\}$ and the finite set $\{\vec{\theta}_M\}$ can be considered as labels $\{M\}$. The difference between this case from the general situation is that the elements of $\{\rho(\vec{Y}|\vec{\theta}_M)\}$ are, nevertheless, not arbitrary distributions, but all of them belong to a given parametric family.

In the general case, we assume that $\rho(\vec{\theta}|M)$ is factorized into a product of two densities: the depending on the object density $\rho(\vec{\mu}|M) = \delta(\vec{\mu}-\vec{\mu}_M)$ and the density $\rho(\vec{\nu})$ that does not depend on any object — it is common for all of them:

$$\rho(\vec{\theta}|M) = \delta(\vec{\mu}-\vec{\mu}_M)\rho(\vec{\nu}), \tag{8.6}$$

and the parameters $\vec{\mu}=\{\mu_1,\ldots,\mu_g\}$, $\vec{\nu}=\{\nu_1,\ldots,\nu_h\}$ are considered as components of the complete system of parameters $\vec{\theta}=\{\vec{\mu},\vec{\nu}\}$,

$r = g + h$. As a result, a complete statistical description of the data (8.5) takes the form of an expectation of the conditional distribution $\rho_M(\vec{Y}|\vec{\theta}) = \rho(\vec{Y}|\vec{\mu}_M, \vec{\nu})$ over the *prior distribution* $\rho(\vec{\nu})$:

$$\rho(\vec{Y}|M) = \iint \rho(\vec{Y}|\vec{\mu}, \vec{\nu})\delta(\vec{\mu} - \vec{\mu}_M)\rho(\vec{\nu})d\vec{\mu}d\vec{\nu}$$

$$= \int \rho(\vec{Y}|\vec{\mu}_M, \vec{\nu})\rho(\vec{\nu})d\vec{\nu} = \rho(\vec{Y}|\vec{\mu}_M). \tag{8.7}$$

Note that in the context of (8.7), parameters $\vec{\mu}_M$ are no longer associated with the given registered precedent with the parameters $\vec{\theta} = \{\vec{\mu}, \vec{\nu}\}$ but are now determined by the whole class of precedents obtained from the given one for other values of the common parameters $\vec{\nu}$.

Based on the definitions and assumptions discussed, it is now possible to formulate the main problem of the approach as follows:

According to the data observed from the object $\vec{X}$ to choose the most matching class of precedents M for the object.

Note that the matching implies statistical consistency between $\vec{X}$ and M, which should correspond to the significance level of the marginal distribution $\rho(\vec{X}|\vec{\mu}_M)$, following from (8.3), (8.7):

$$\rho(\vec{X}|\vec{\mu}_M) = \int \rho(\vec{Y}|\vec{\mu}_M)d\vec{Z} = \int \left\{ \int \rho(\vec{Y}|\vec{\mu}_M, \vec{\nu})\rho(\vec{\nu})d\vec{\nu} \right\} d\vec{Z}$$

$$= \int \left\{ \int \rho(\vec{X}, \vec{Z}|\vec{\mu}_M, \vec{\nu})d\vec{Z} \right\} \rho(\vec{\nu})d\vec{\nu}$$

$$= \int \rho(\vec{X}|\vec{\mu}_M, \vec{\nu})\rho(\vec{\nu})d\vec{\nu}. \tag{8.8}$$

As it follows from (8.8), the best matching class M for a given data $\vec{X}$ is the one that gives the largest value of $\rho(\vec{X}|\vec{\mu}_M)$ with respect to $\vec{\mu}_M$. This notification can be formalized in the form of Fisher's maximum likelihood (ML) principle, namely, ML proposes to estimate the target class with the label M_{ML}, which ensures the ML function $\rho(\vec{X}|\vec{\mu}_M)$ (8.8) on the labels set $\{M\}$:

$$M_{\mathrm{ML}} = arg \max_{\{M\}}(\rho(\vec{X}|\vec{\mu}_M)). \tag{8.9}$$

In the case when the correspondence $M \to \vec{\mu}_M$ is not initially specified, it should be restored beforehand based on a training process — registering and describing data $\vec{X}_M$ for objects that make up

the class M. For these purposes, the ML principle can also be used, but the maximization, in contrast to (8.9), should be carried out for a continuous set of parameters $\{\vec{\mu}\}$:

$$\vec{\mu}_M = arg \max_{\{\vec{\mu}\}} (\rho(\vec{X}_M|\vec{\mu})). \tag{8.10}$$

The division of the above formulated identification problem into tasks of learning (training) (8.10) and recognition (testing) (8.9) is the main paradigm of machine learning [8]. The following discussion is almost entirely devoted to the synthesis of such procedures using the statistical model (8.7) and some variants of EM-like algorithms.

Note that despite the conceptual simplicity of problems (8.9) and (8.10), their practical implementation often encounters a number of technical problems. Some of them can be bypassed by replacing the likelihood function $\rho(\vec{X}_M|\vec{\mu})$ by its logarithm $L(\vec{X}_M;\vec{\mu}) = \ln \rho(\vec{X}_M|\vec{\mu})$. Therefore, further in the work we use the logarithmic likelihood functions.

8.3.2 *EM-like algorithm for learning procedure*

According to (8.10), the most general form of the learning (training) problem is formulated as follows: define for any arbitrary data/label pair $\{(\vec{X}_M, M)\}$ the corresponding ML parameters $\vec{\mu}_M$:

$$\begin{aligned} \vec{\mu}_M &= arg \max_{\{\vec{\mu}\}} (L(\vec{X}_M;\vec{\mu})), \\ L(\vec{X}_M;\vec{\mu}) &= \ln \rho(\vec{X}_M|\vec{\mu}). \end{aligned} \tag{8.11}$$

A direct solution to (8.11) can lead to significant difficulties due to the summation over $\vec{\nu}$ in (8.8) when finding $\rho(\vec{X}_M|\vec{\mu})$. To get around this problem, it may be useful to approximate the log-likelihood function $L(\vec{X}_M;\vec{\mu})$ (8.11) by the extended logarithmic function $L(\vec{X}_M, \vec{Z};\vec{\mu},\vec{\nu})$, as the latter is sometimes easier to calculate. The simplest decision here is the linear approximation $\hat{L}(\vec{X}_M;\vec{\mu})$:

$$\begin{aligned} L(\vec{X}_M;\vec{\mu}) &\sim \hat{L}(\vec{X}_M;\vec{\mu}) \\ &= \iint k(\vec{X}_M,\vec{Z};\vec{\nu}) L(\vec{X}_M,\vec{Z};\vec{\mu},\vec{\nu}) d\vec{Z} d\vec{\nu} + b(\vec{X}_M), \\ L(\vec{X}_M,\vec{Z};\vec{\mu},\vec{\nu}) &= \ln \rho(\vec{X}_M,\vec{Z}|\vec{\mu},\vec{\nu}), \end{aligned} \tag{8.12}$$

where $b(\vec{X}_M)$ is some additive, independent of $\vec{\mu}$ term and $k(\vec{X}_M, \vec{Z}; \vec{\nu})$ is some integral kernel, also independent of $\vec{\mu}$. If we note that the extended log-likelihood function can be expressed in terms of the original one by

$$\begin{aligned}\ln \rho(\vec{X}_M, \vec{Z}|\vec{\mu}, \vec{\nu}) &= \ln[\rho(\vec{Z}; \vec{\nu}|\vec{X}_M; \vec{\mu})\rho(\vec{X}_M; \vec{\mu})/\rho(\vec{\mu}, \vec{\nu})] \\ &= \ln[\rho(\vec{Z}; \vec{\nu}|\vec{X}_M; \vec{\mu})/\rho(\vec{\nu})] + L(\vec{X}_M; \vec{\mu}), \quad (8.13)\end{aligned}$$

then for the approximation error $\delta(\vec{X}_M; \vec{\mu})$ we can write the following:

$$\begin{aligned}\delta(\vec{X}_M; \vec{\mu}) &= L(\vec{X}_M; \vec{\mu}) - \hat{L}(\vec{X}_M; \vec{\mu}) \\ &= L(\vec{X}_M; \vec{\mu}) \left[1 - \iint k(\vec{X}_M, \vec{Z}; \vec{\nu}) d\vec{Z} d\vec{\nu}\right] \\ &\quad - \left[\iint k(\vec{X}_M, \vec{Z}; \vec{\nu}) \ln[\rho(\vec{Z}; \vec{\nu}|\vec{X}_M; \vec{\mu})/\rho(\vec{\nu})] d\vec{Z} d\vec{\nu} + b(\vec{X}_M)\right]. \quad (8.14)\end{aligned}$$

Of course, in the general case, no choice of $b(\vec{X}_M)$ and kernel $k(\vec{X}_M, \vec{Z}; \vec{\nu})$ can identically with respect to estimated $\vec{\mu}$ turn the right-hand side of (8.14) to zero. Nevertheless, we can turn it to zero for some $\vec{\mu}^{(*)}$ if we impose a normalization condition on $k(\vec{X}_M, \vec{Z}; \vec{\nu})$ for any $\vec{X}_M$ and set $b(\vec{X}_M)$ equal to

$$b(\vec{X}_M) = -\iint k(\vec{X}_M, \vec{Z}; \vec{\nu}) \ln[\rho(\vec{Z}; \vec{\nu}|\vec{X}_M; \vec{\mu}^{(*)})/\rho(\vec{\nu})] d\vec{Z} d\vec{\nu}.$$

The expression for approximation error (8.14) with this choice takes the following form:

$$\delta^{(*)}(\vec{X}_M; \vec{\mu}) = -\iint k(\vec{X}_M, \vec{Z}; \vec{\nu}) \ln\left[\frac{\rho(\vec{Z}; \vec{\nu}|\vec{X}_M; \vec{\mu})}{\rho(\vec{Z}; \vec{\nu}|\vec{X}_M; \vec{\mu}^{(*)})}\right] d\vec{Z} d\vec{\nu}. \quad (8.15)$$

If we set the kernel $k(\vec{X}_M; \vec{Z}, \vec{\nu})$ equal to the density $\rho(\vec{Z}; \vec{\nu}|\vec{X}_M; \vec{\mu}^{(*)})$, then the expression for the error $\delta^{(*)}(\vec{X}_M; \vec{\mu})$ becomes the Kullback–Leibler divergence between $\rho(\vec{Z}; \vec{\nu}|\vec{X}_M; \vec{\mu})$ and $\rho(\vec{Z}; \vec{\nu}|\vec{X}_M; \vec{\mu}^{(*)})$. Since the latter is quadratic in the variation $\delta\vec{\mu} =$

$\vec{\mu} - \vec{\mu}^{(*)}$, the approximation $\hat{L}(\vec{X}_M; \vec{\mu})$ (8.12) is not only equal but also tangent to $L(\vec{X}_M; \vec{\mu})$ at $\vec{\mu}^{(*)}$.

Thus, the tangent to $L(\vec{X}_M; \vec{\mu})$ at some point $\vec{\mu}^{(*)}$ approximation $\hat{L}^{(*)}(\vec{X}_M; \vec{\mu})$ and the corresponding approximation error are as follows:

$$\begin{aligned}\hat{L}^{(*)}(\vec{X}_M; \vec{\mu}) &= \iint \rho(\vec{Z}; \vec{\nu}|\vec{X}_M; \vec{\mu}^{(*)}) \ln \left[\frac{\rho(\vec{X}_M, \vec{Z}; \vec{\nu}|\vec{\mu})}{\rho(\vec{Z}; \vec{\nu}|\vec{X}_M; \vec{\mu}^{(*)})} \right] d\vec{Z} d\vec{\nu} \\ &= L(\vec{X}_M; \vec{\mu}) - \delta(\vec{X}_M; \vec{\mu}), \\ &\quad \delta^{(*)}(\vec{X}_M; \vec{\mu}) = D_{KL}(\vec{X}_M; \vec{\mu}, \vec{\mu}^{(*)}) \\ &= - \iint \rho(\vec{Z}; \vec{\nu}|\vec{X}_M; \vec{\mu}^{(*)}) \ln \left[\frac{\rho(\vec{Z}; \vec{\nu}|\vec{X}_M; \vec{\mu})}{\rho(\vec{Z}; \vec{\nu}|\vec{X}_M; \vec{\mu}^{(*)})} \right] d\vec{Z} d\vec{\nu},\end{aligned} \tag{8.16}$$

where in the expression for $\delta^{(*)}(\vec{X}_M; \vec{\mu})$, it is explicitly emphasized that the error exactly coincides with the Kullback–Leibler divergence $D_{KL}(\vec{X}_M; \vec{\mu}, \vec{\mu}^{(*)})$ between *a posteriori* distributions of hidden data $\vec{Z}$ and parameters $\vec{\nu}$.

Since the Kullback–Leibler divergence is always non-negative and vanishes only when the arguments coincide, the approximation $\hat{L}^{(*)}(\vec{X}_M; \vec{\mu})$ has other remarkable properties, for example, it is the strict lower bound of $L(\vec{X}_M; \vec{\mu})$, coinciding with it at a single point $\vec{\mu} = \vec{\mu}^{(*)}$.

Bearing in mind the listed properties of $\hat{L}^{(*)}(\vec{X}_M; \vec{\mu})$, let us consider the set of problems like (8.11) but for all possible approximations $\hat{L}^{(*)}(\vec{X}_M; \vec{\mu})$:

$$\vec{\mu}^{(**)} = arg \max_{\{\vec{\mu}\}} (\hat{L}^{(*)}(\vec{X}_M; \vec{\mu})). \tag{8.17}$$

All solutions $\vec{\mu}^{(**)}$ for all these problems define the depending on $\vec{X}_M$ parameter mapping $T(\vec{\mu}^{(*)}) = \vec{\mu}^{(**)}$. This mapping $T : \{\vec{\mu}\} \to \{\vec{\mu}\}$ is obviously monotonous in the following sense:

$$L(\vec{X}_M; \vec{\mu}^{(*)}) = \hat{L}^{(*)}(\vec{X}_M; \vec{\mu}^{(*)}) \le \hat{L}^{(*)}(\vec{X}_M; \vec{\mu}^{(**)}) \le L(\vec{X}_M; \vec{\mu}^{(**)}). \tag{8.18}$$

Note that in the case of $\vec{\mu}^{(*)} \neq \vec{\mu}^{(**)}$ the strict inequalities hold in (8.18). Indeed, if, for example, $\hat{L}^{(*)}(\vec{X}_M; \vec{\mu}^{(**)}) = L(\vec{X}_M; \vec{\mu}^{(**)})$, then,

according to the above-listed properties of $\hat{L}^{(*)}(\vec{X}_M;\vec{\mu})$, it would follow that $\vec{\mu}^{(*)} = \vec{\mu}^{(**)}$ — contradiction. On the contrary, in the case of $\vec{\mu}^{(*)} = \vec{\mu}^{(**)}$, only equal signs will take place in (8.18). Moreover, in this case, $\vec{\mu}^{(*)}$ is a stationary point of mapping T: $T(\vec{\mu}^{(*)}) = \vec{\mu}^{(*)}$, and in this stationary point, $\hat{L}^{(*)}(\vec{X}_M;\vec{\mu})$ and $L(\vec{X}_M;\vec{\mu})$ both reach a (local) maximum, due to their tangency in $\vec{\mu}^{(*)}$. Thus, all maxima of $\hat{L}^{(*)}(\vec{X}_M;\vec{\mu})$ and $L(\vec{X}_M;\vec{\mu})$ coincide. In particular, the global maximum of $L(\vec{X}_M;\vec{\mu})$ — the solution $\vec{\mu}_M$ of problem (8.11) — coincides with one of the $\hat{L}^{(*)}(\vec{X}_M;\vec{\mu})$ maxima.

To find the stationary points of the mapping T, the standard technique for recurrent calculations can be used, namely, choosing some initial estimate $\vec{\mu}^{(0)}$, we can construct successive T-iterations: $\vec{\mu}^{(\iota)} = T(\vec{\mu}^{(\iota-1)})$, $\iota = 1, 2, \ldots$. If the sequence $\vec{\mu}^{(0)}, \vec{\mu}^{(1)}, \ldots$ converges to the limit $\vec{\mu}^{(*)}$, then $\vec{\mu}^{(*)} = T(\vec{\mu}^{(*)})$ takes place, and $\vec{\mu}^{(*)}$ is necessarily the stationary (fixed) point.

The technique described above for finding stationary points of the mapping $T : \{\vec{\mu}\} \to \{\vec{\mu}\}$ underlies the EM algorithm for finding local solutions to problem (8.11), namely, the EM algorithm calculates successive refinements $\vec{\mu}^{(\iota)}$ to the limit solution $\vec{\mu}^{(*)}$, recurrently performing the following two-step procedure:

E-step: According to the refinement $\vec{\mu}^{(\iota-1)}$ available at iteration (ι), the Q-function — dependent on the $\vec{\mu}$ part of $\hat{L}^{(\iota-1)}(\vec{X}_M;\vec{\mu})$ (8.16) — is calculated. This Q-function is essentially an expectation of the extended log-likelihood function $L(\vec{X}_M, \vec{Z};\vec{\mu},\vec{\nu})$ (8.12) over the *posteriori* distribution $\rho(\vec{Z};\vec{\nu}|\vec{X}_M;\vec{\mu}^{(*)})$ of hidden data and parameters $\vec{\nu}$:

$$Q(\vec{X}_M;\vec{\mu},\vec{\mu}^{(\iota-1)}) = \iint \rho(\vec{Z};\vec{\nu}|\vec{X}_M;\vec{\mu}^{(\iota-1)}) \ln[\rho(\vec{X}_M,\vec{Z}|\vec{\mu},\vec{\nu})] d\vec{Z} d\vec{\nu}. \quad (8.19)$$

M-step: The maximum point $\vec{\mu}^{(\iota)}$ of the Q-function is calculated:

$$\vec{\mu}^{(\iota)} = arg \max_{\{\vec{\mu}\}} (Q(\vec{X}_M;\vec{\mu},\vec{\mu}^{(\iota-1)})). \quad (8.20)$$

If the stopping algorithm condition is satisfied, for example, $\delta^{(\iota-1)}(\vec{X}_M;\vec{\mu}^{(\iota)}) = D_{KL}(\vec{X}_M;\vec{\mu}^{(\iota)},\vec{\mu}^{(\iota-1)}) < \varepsilon$, then $\vec{\mu}_M = \vec{\mu}^{(\iota)}$ is accepted and the EM algorithm is completed, if not, then $\vec{\mu}^{(\iota)}$ is accepted as the current refinement of the solution $\vec{\mu}_M$ and the algorithm proceeds to **E-step**.

Note that if the *a posteriori* distribution $\rho(\vec{\nu}|\vec{X}_M;\vec{\mu}^{(\iota-1)})$ has the form of a narrow peak in the vicinity of some point $\vec{\nu}^{(\iota-1)} = arg\max(\rho(\vec{\nu}|\vec{X}_M;\vec{\mu}^{(\iota-1)}))$, then we can approximately write the following:

$$\rho(\vec{Z};\vec{\nu}|\vec{X}_M;\vec{\mu}^{(\iota-1)}) \approx \rho(\vec{Z}|\vec{X}_M;\vec{\mu}^{(\iota-1)},\vec{\nu}^{(\iota-1)})\rho(\vec{\nu}|\vec{X}_M;\vec{\mu}^{(\iota-1)}),$$

and we obtain the following approximation for the Q-function, which somewhat simplifies the calculations of the EM algorithm:

$$Q(\vec{X}_M;\vec{\mu},\vec{\mu}^{(\iota-1)},\vec{\nu}^{(\iota-1)}) = \int \rho(\vec{Z}|\vec{X}_M;\vec{\mu}^{(\iota-1)},\vec{\nu}^{(\iota-1)})\ln[\rho(\vec{X}_M,\vec{Z}|\vec{\mu},\vec{\nu}^{(\iota-1)})]d\vec{Z}. \quad (8.21)$$

It is worth noting that the maximum point $\vec{\nu}^{(\iota-1)}$ in (8.21) can also be found using the internal EM algorithm, namely, if we use the Bayes formula

$$\rho(\vec{\nu}|\vec{X}_M;\vec{\mu}^{(\iota-1)}) = \frac{\rho(\vec{X}_M|\vec{\mu}^{(\iota-1)},\vec{\nu})}{\rho(\vec{X}_M|\vec{\mu}^{(\iota-1)})}\rho(\vec{\nu}),$$

and assume a wide *a priori* distribution $\rho(\vec{\nu})$ then $\vec{\nu}^{(\iota-1)}$ can be approximated by the maximum point of the distribution $\rho(\vec{X}_M|\vec{\mu}^{(\iota-1)},\vec{\nu})$, which is an ML estimator for the log-likelihood $l(\vec{X}_M,\vec{\mu}^{(\iota-1)},\vec{\nu}) = \ln\rho(\vec{X}_M|\vec{\mu}^{(\iota-1)},\vec{\nu})$:

$$\vec{\nu}^{(\iota-1)} = arg\max_{\{\vec{\nu}\}}(l(\vec{X}_M,\vec{\mu}^{(\iota-1)},\vec{\nu})). \quad (8.22)$$

If, as described above, we apply for $l(\vec{X}_M,\vec{\mu}^{(\iota-1)},\vec{\nu})$ the best linear approximation

$$\hat{l}^{(*)}(\vec{X}_M,\vec{\mu}^{(\iota-1)},\vec{\nu}) = \int \rho(\vec{Z}|\vec{X}_M;\vec{\mu}^{(\iota-1)},\vec{\nu}^{(*)})\ln\left[\frac{\rho(\vec{X}_M,\vec{Z}|\vec{\mu}^{(\iota-1)},\vec{\nu})}{\rho(\vec{Z}|\vec{X}_M;\vec{\mu}^{(\iota-1)},\vec{\nu}^{(*)})}\right]d\vec{Z}, \quad (8.23)$$

then an internal with respect to the main EM algorithm (8.19)–(8.20) the E′M′ algorithm can be used to find $\vec{\nu}^{(\iota-1)}$ (8.22):

E′-step: According to the refinement $\vec{\nu}^{(\kappa-1)}$ available at iteration (κ), the q-function — dependent on $\vec{\nu}$ part of $\hat{l}^{(*)}(\vec{X}_M, \vec{\mu}^{(\iota-1)}, \vec{\nu})$ (8.23) — is constructed:

$$q(\vec{X}_M; \vec{\mu}^{(\iota-1)}, \vec{\nu}, \vec{\nu}^{(\kappa-1)}) = \int \rho(\vec{Z}|\vec{X}_M; \vec{\mu}^{(\iota-1)}, \vec{\nu}^{(\kappa-1)}) \ln[\rho(\vec{X}_M, \vec{Z}|\vec{\mu}^{(\iota-1)}, \vec{\nu})] d\vec{Z}. \quad (8.24)$$

M′-step: The maximum point $\vec{\nu}^{(\kappa)}$ of the q-function is calculated:

$$\vec{\nu}^{(\kappa)} = arg \max_{\{\vec{\nu}\}} (q(\vec{X}_M; \vec{\mu}^{(\iota-1)}, \vec{\nu}, \vec{\nu}^{(\kappa-1)})). \quad (8.25)$$

The stopping condition can be set, for example, as $\delta^{(\kappa)}(\vec{X}_M; \vec{\nu}^{(\kappa)}) < \varepsilon$, where

$$\begin{aligned} \delta^{(\kappa)}(\vec{X}_M; \vec{\nu}^{(\kappa)}) &= D_{KL}(\vec{X}_M; \vec{\mu}^{(\iota-1)}, \vec{\nu}^{(\kappa)}, \vec{\nu}^{(\kappa-1)}) \\ &= \int \rho(\vec{Z}|\vec{X}_M; \vec{\mu}^{(\iota-1)}, \vec{\nu}^{(\kappa-1)}) \\ &\quad \times \ln \left[\frac{\rho(\vec{Z}|\vec{X}_M; \vec{\mu}^{(\iota-1)}, \vec{\nu}^{(\kappa)})}{\rho(\vec{Z}|\vec{X}_M; \vec{\mu}^{(\iota-1)}, \vec{\nu}^{(\kappa-1)})} \right] d\vec{Z} \end{aligned} \quad (8.26)$$

is the Kullback–Leibler divergence between the *a posteriori* distributions of the hidden data $\vec{Z}$ for the given parameters $\vec{\mu}^{(\iota-1)}, \vec{\nu}^{(\kappa-1)}$, and $\vec{\nu}^{(\kappa)}$.

8.3.3 *EM-like algorithm for identification procedure*

According to (8.9), the problem of identification (testing) is formulated as the following problem of discrete maximization:

$$\begin{aligned} M_{\mathrm{ML}} &= arg \max_{M} (L(\vec{X}; \vec{\mu}_M)), \\ L(\vec{X}; \vec{\mu}_M) &= \ln \rho(\vec{X}|\vec{\mu}_M). \end{aligned} \quad (8.27)$$

Note that although the set $\{M\}$ is finite and problem (8.27) can be solved in a finite number of steps, in some cases, as well as in training, it is advisable to use specialized identification procedures that are similar in content to the EM algorithm.

Based on considerations repeating those given in the previous section, we can also construct for the logarithmic likelihood function $L(\vec{X};\vec{\mu}_M)$ (8.27) an approximation $\hat{L}^{(N)}(\vec{X};\vec{\mu}_M)$, which coincides with it up to the positive Kullback–Leibler divergence of posterior distributions $\rho(\vec{Z};\vec{\nu}|\vec{X};\vec{\mu}_M)$ and $\rho(\vec{Z};\vec{\nu}|\vec{X};\vec{\mu}_N)$:

$$\begin{aligned}\hat{L}^{(N)}(\vec{X};\vec{\mu}_M) &= \iint \rho(\vec{Z};\vec{\nu}|\vec{X};\vec{\mu}_N)\ln\left[\frac{\rho(\vec{X},\vec{Z};\vec{\nu}|\vec{\mu}_M)}{\rho(\vec{Z};\vec{\nu}|\vec{X};\vec{\mu}_N)}\right]d\vec{Z}d\vec{\nu}\\ &= L(\vec{X};\vec{\mu}_M) - \delta^{(N)}(\vec{X};\vec{\mu}_M);\\ \delta^{(N)}(\vec{X};\vec{\mu}_M) &= D_{KL}(\vec{X};\vec{\mu}_M,\vec{\mu}_N)\\ &= -\iint \rho(\vec{Z};\vec{\nu}|\vec{X};\vec{\mu}_N)\ln\left[\frac{\rho(\vec{Z};\vec{\nu}|\vec{X};\vec{\mu}_M)}{\rho(\vec{Z};\vec{\nu}|\vec{X};\vec{\mu}_N)}\right]d\vec{Z}d\vec{\nu}.\end{aligned} \tag{8.28}$$

Let us denote by $N^{(*)}$ the solution of problem (8.27), where instead of $L(\vec{X};\vec{\mu}_M)$ we have $\hat{L}^{(N)}(\vec{X};\vec{\mu}_M)$ (if there are several solutions, but there is no $\vec{\mu}_N$ among them, we will choose any, if there is, then we will choose it). This solution as before defines the mapping $T:\{M\}\to\{M\}$ with the only difference that $\{M\}$ is now a finite set:

$$T(N) = N^{(*)} = arg\max_M(\hat{L}^{(N)}(\vec{X};\vec{\mu}_M)). \tag{8.29}$$

It is easy to show that this map also has the monotonicity property:

$$L(\vec{X};\vec{\mu}_N) = \hat{L}^{(N)}(\vec{X};\vec{\mu}_N) \le \hat{L}^{(N)}(\vec{X};\vec{\mu}_{N^{(*)}}) \le L(\vec{X};\vec{\mu}_{N^{(*)}}). \tag{8.30}$$

Note that for $N \neq N^{(*)}$ the strict inequalities hold in (8.30). In the opposite case, $N^{(*)}$ is a stationary point of mapping T (8.29).

Note that the global maximum of $L(\vec{X}_M;\vec{\mu})$ — the solution M_{ML} of the problem (8.27) — necessarily coincides with one of the stationary points of mapping T (8.29), otherwise in point $M^{(*)}_{ML} = T(M_{ML})$, $M^{(*)}_{ML} \neq M_{ML}$, the value of the log-likelihood function $L(\vec{X};M^{(*)}_{ML})$ will be greater than the maximum $L(\vec{X};M_{ML})$.

To find the stationary points of the mapping T (8.29), it is possible to use the standard technique for recurrent calculations, namely, by

choosing some initial estimate $M^{(0)}$, we can construct the successive of T-iterations: $M^{(\iota)} = T(M^{(\iota-1)})$, $\iota = 1, 2, \ldots$. In the sequence $M^{(0)}, M^{(1)}, \ldots$, due to the finiteness of the set $\{M\}$, starting with a certain number $\iota^{(*)}$ the value $M^{(*)}$ will repeat itself (in view of (8.30), there cannot be cycles). This value $M^{(*)}$ is, by definition, the stationary point of T. So, a stationary point can be always found in finite number of iterations.

The above method for finding stationary points of the map T (8.29) is the basis of the EM algorithm for finding solution to problem (8.27). It consists of the recurrent execution of the following two-step procedure:

E-step: According to the parameters $\vec{\mu}_N$ of class N, available at iteration (ι), the $\vec{\mu}_M$-dependent part of $\hat{L}^{(N)}(\vec{X}; \vec{\mu}_M)$ (8.28) — Q-function — is constructed:

$$Q(\vec{X}; \vec{\mu}_M, \vec{\mu}_N) = \iint \rho(\vec{Z}; \vec{\nu}|\vec{X}; \vec{\mu}_N) \ln[\rho(\vec{X}, \vec{Z}|\vec{\mu}_M, \vec{\nu})] d\vec{Z} d\vec{\nu}. \quad (8.31)$$

M-step: Looking up for the class $N^{(*)}$ that corresponds to parameters $\vec{\mu}_{N^{(*)}}$ maximizing the Q-function:

$$\vec{\mu}_{N^{(*)}} = arg \max_M (Q(\vec{X}; \vec{\mu}_M, \vec{\mu}_N)). \quad (8.32)$$

If $N^{(*)} = N$, then it is accepted as solution $M^{(*)}$ and the algorithm terminates, if not, then $N = N^{(*)}$ is accepted as the current refinement of the $M^{(*)}$ and the algorithm proceeds to **E-step**.

Note that if the *a posteriori* distribution $\rho(\vec{\nu}|\vec{X}; \vec{\mu}_N)$ has a narrow peak in the vicinity of the point $\vec{\nu}_N = arg \max(\rho(\vec{\nu}|\vec{X}; \vec{\mu}_N))$, then rewriting $\rho(\vec{Z}; \vec{\nu}|\vec{X}; \vec{\mu}_N)$ decomposition as in the previous section

$$\rho(\vec{Z}; \vec{\nu}|\vec{X}; \vec{\mu}_N) \approx \rho(\vec{Z}|\vec{X}; \vec{\mu}_N, \vec{\nu}_N) \rho(\vec{\nu}|\vec{X}; \vec{\mu}_N),$$

we get the following Q-function approximation, which somewhat simplifies the calculations of the EM algorithm:

$$Q(\vec{X}; \vec{\mu}_M, \vec{\mu}_N, \vec{\nu}_N) \approx \int \rho(\vec{Z}|\vec{X}; \vec{\mu}_N, \vec{\nu}_N) \ln[\rho(\vec{X}, \vec{Z}|\vec{\mu}_M, \vec{\nu}_N)] d\vec{Z}. \quad (8.33)$$

The maximum point $\vec{\nu}^{(\iota-1)}$ in (8.33) can also be found using the internal EM algorithm, namely, if we use the Bayes formula

$$\rho(\vec{\nu}|\vec{X};\vec{\mu}_N) = \frac{\rho(\vec{X}|\vec{\mu}_N,\vec{\nu})}{\rho(\vec{X}|\vec{\mu}_N)}\rho(\vec{\nu})$$

and assume a wide *a priori* distribution $\rho(\vec{\nu})$ then $\vec{\nu}_N$ can be approximated as the point of the distribution $\rho(\vec{X}|\vec{\mu}_N,\vec{\nu})$ maximum, which is exactly ML estimate for the log-likelihood function $l(\vec{X};\vec{\mu}_N,\vec{\nu}) = \ln\rho(\vec{X}|\vec{\mu}_N,\vec{\nu})$:

$$\vec{\nu}_N = arg\max_{\vec{\nu}}(l(\vec{X};\vec{\mu}_N,\vec{\nu})). \tag{8.34}$$

If we apply for $l(\vec{X};\vec{\mu}_N,\vec{\nu})$ the best linear approximation

$$\hat{l}^{(*)}(\vec{X};\vec{\mu}_N,\vec{\nu}) \approx \int \rho(\vec{Z}|\vec{X};\vec{\mu}_N,\vec{\nu}^{(*)})\ln\left[\frac{\rho(\vec{X},\vec{Z}|\vec{\mu}_N,\vec{\nu})}{\rho(\vec{Z}|\vec{X};\vec{\mu}_N,\vec{\nu}^{(*)})}\right]d\vec{Z}, \tag{8.35}$$

then the internal E′M′ algorithm can be used to find $\vec{\nu}_N$ (8.34):

E′-step: According to the refinement $\vec{\nu}_N^{(\kappa-1)}$ available at iteration (κ), the q-function – dependent on $\vec{\nu}$ part of $\hat{l}^{(\kappa-1)}(\vec{X};\vec{\mu}_N,\vec{\nu})$ (8.35) — is constructed:

$$q(\vec{X};\vec{\mu}_N,\vec{\nu},\vec{\nu}_N^{(\kappa-1)}) = \int \rho(\vec{Z}|\vec{X};\vec{\mu}_N,\vec{\nu}_N^{(\kappa-1)})\ln[\rho(\vec{X},\vec{Z}|\vec{\mu}_N,\vec{\nu})]d\vec{Z}. \tag{8.36}$$

M′-step: The maximum point $\vec{\nu}^{(\kappa)}$ of the q-function (8.36) is calculated:

$$\vec{\nu}_N^{(\kappa)} = arg\max_{\{\vec{\nu}\}}(q(\vec{X};\vec{\mu}_N,\vec{\nu},\vec{\nu}_N^{(\kappa-1)})). \tag{8.37}$$

If the stopping condition is satisfied, for example, $\delta^{(\kappa)}(\vec{X};\vec{\mu}_N,\vec{\nu}_N^{(\kappa)}) < \varepsilon$, where

$$\delta^{(\kappa)}(\vec{X};\vec{\mu}_N,\vec{\nu}_N^{(\kappa)}) = D_{KL}(\vec{X};\vec{\mu}_N,\vec{\nu}_N^{(\kappa)},\vec{\nu}_N^{(\kappa-1)}) = -\int \rho(\vec{Z}|\vec{X};\vec{\mu}_N,\vec{\nu}_N^{(\kappa-1)})\ln\left[\frac{\rho(\vec{Z}|\vec{X};\vec{\mu}_N,\vec{\nu}_N^{(\kappa)})}{\rho(\vec{Z}|\vec{X};\vec{\mu}_N,\vec{\nu}_N^{(\kappa-1)})}\right]d\vec{Z}, \tag{8.38}$$

then $\vec{\nu}_N = \vec{\nu}_N^{(\kappa)}$ is accepted and the E′M′ algorithm terminates, if not, then $\vec{\nu}_N^{(\kappa)}$ is accepted as the current refinement of the solution $\vec{\nu}_N$ and the algorithm proceeds to step E′ to execute the next iteration.

8.4 Target Shape Identification in Gaussian Mixtures

The previous section presents the general statistical framework for the identification problems. The proposed approach is essentially based on the principles of machine learning. However, the great popularity of machine learning methods is associated not only with their progressive concept but also with the focus on the well-structured computational procedures.

In order to give the general statistical approach the form of computational procedures and algorithms, it is necessary to refine the models of the problem being solved. As follows from the previous section, first of all, it is necessary to clarify the complete descriptions $\rho(\vec{Y}|\vec{\theta})$ (8.5) in the form of a parametric family of complete data $\vec{Y} \sim \{\vec{X}, \vec{Z}\}$ probability distributions.

The observed (registered) data $\vec{X}$, as it is described in Section 8.2, are given in the form of a discrete set of count coordinates $\{\vec{x}_1, \vec{x}_2, \ldots, \vec{x}_n\}$. They can be considered as a random realization of some PPP. In the context of the identification by the form (and not by brightness) of objects, we can assume that the number of counts n is given and, therefore, all coordinates $\vec{x}_i$ are independent and equally distributed on the sensor-sensitive surface Ω [12]. Thus, the main feature of the refined model is the factorization of the observed data distribution into the product of the densities of individual counts:

$$\begin{aligned} \rho(\vec{X}|\vec{\theta}) &= \rho(\{\vec{x}_i\}|\vec{\mu},\vec{\nu}) = \prod_{i=1}^{n} \rho(\vec{x}_i|\vec{\mu},\vec{\nu}) \\ \rho(\vec{x}|\vec{\mu},\vec{\nu}) &= \frac{I(\vec{x}|\vec{\mu},\vec{\nu})}{\int_\Omega I(\vec{x}|\vec{\mu},\vec{\nu})d\vec{x}}. \end{aligned} \tag{8.39}$$

If we assume that the intensity $I(\vec{x}|\vec{\mu},\vec{\nu})$ of the incident on Ω radiation can be represented by the sum of K incoherent components

$$I(\vec{x}|\vec{\mu},\vec{\nu}) = \sum_{z=1}^{K} I_z(\vec{x}|\vec{\mu},\vec{\nu}),$$

then we get

$$\rho(\vec{x}|\vec{\mu},\vec{\nu}) = \sum_{z=1}^{K} p_z i_z(\vec{x}|\vec{\mu},\vec{\nu}) = \sum_{z=1}^{K} p_z \rho(\vec{x}|z;\vec{\mu},\vec{\nu})$$

$$p_z = \frac{\int_\Omega I_z(\vec{x}|\vec{\mu},\vec{\nu})d\vec{x}}{\int_\Omega I(\vec{x}|\vec{\mu},\vec{\nu})d\vec{x}}, i_z(\vec{x}|\vec{\mu},\vec{\nu}) = \frac{I_z(\vec{x}|\vec{\mu},\vec{\nu})}{\int_\Omega I_z(\vec{x}|\vec{\mu},\vec{\nu})d\vec{x}} = \rho(\vec{x}|z;\vec{\mu},\vec{\nu}), \tag{8.40}$$

where the normalized component intensities $i_z(\vec{x}|\vec{\mu},\vec{\nu}) = \rho(\vec{x}|z;\vec{\mu},\vec{\nu})$ are presented as explicitly parametrized by $\vec{\mu},\vec{\nu}$, whose meaning is concretized later on. It is easy to notice that the marginal distribution of observed data $\vec{x}$ (8.40) is a mixture of conditional densities, so it is natural to accept component indices z as the hidden indicators for independent counts $\vec{x}_i$. So, the joint distribution of $\vec{x}$ and z has the following form:

$$\rho(\vec{x},z|\vec{\mu},\vec{\nu}) = p_z \rho(\vec{x}|z;\vec{\mu},\vec{\nu}). \tag{8.41}$$

If we assume that indicators z_i for independent counts $\vec{x}_i$ are also independent, it immediately results in factorization of the complete data distribution:

$$\rho(\vec{X},\vec{Z}|\vec{\theta}) = \rho(\{\vec{x}_i,z_i\}|\vec{\mu},\vec{\nu}) = \prod_{i=1}^{n} p_{z_i}\rho(\vec{x}_i|z_i;\vec{\mu},\vec{\nu}). \tag{8.42}$$

Thus, the above analysis can be considered as the choice of the structure of complete statistical descriptions $\rho(\vec{Y}|\vec{\theta})$. The mentioned assumption that all hidden $\{z_i\}$ are also iid random values can be considered as the second model specification.

The third and last specification concerns the types of conditional distribution densities of mixtures (8.42). A flexible tool for modeling a wide class of multivariable distributions is a family of Gaussian mixture models (GMM) [11]. GMM components are well-known normal distributions $N(\vec{x}|\vec{m},\hat{R})$ with parameters $\vec{m}$ (mean) and $\hat{R}$ (correlation matrix of distribution). These parameters allow a transparent interpretation in terms of the geometric characteristics of the intensity components: $\vec{m}$ is the position of the component center and $\hat{R}$ describes component characteristic scales. Thus, the set of parameters $\vec{\theta}$ that specify the K-component GMM can be given by the

set of triples $\{(p_z, \vec{m}_z, A_z = \hat{R}_z^{-1})\}$, and they are usually sufficient for approximating the intensities of objects having a rather complex shape.

The choice of the GMM in addition to the convenience of modeling/analyzing is also useful because it is an ideal subject for the above-introduced EM algorithms [7]. The remaining part of the chapter is devoted to the issues of algorithmic implementation of the approach proposed, based mainly on GMM models.

8.4.1 *Gaussian mixtures and their transformation classes*

In this section, we recall some definitions concerning GGM and describe the relation between the general parameters $\vec{\theta}$ of GGM and the complete system of precedent parameters $\{\vec{\mu}, \vec{\nu}\}$ in the problem under consideration.

In the GGM, the complete data $\vec{y} = \{\vec{x}, z\}$ distribution is given by the finite probability distribution $\{p_z\}$, $z = 1, \ldots, K$, of (hidden) component indicator z and conditional normal distribution $\rho(\vec{x}|z) = N(\vec{x}|\vec{m}_z, \hat{R}_z)$ of the observed data $\vec{x}$:

$$\rho(\vec{y}|\vec{\theta}) = \rho(\vec{x}, z|\vec{\theta}) = p_z N(\vec{x}|\vec{m}_z, \hat{R}_z), \tag{8.43}$$

where $\vec{\theta} = \{(p_z, \vec{m}_z, \hat{A}_z)\}$, $\hat{A}_z = \hat{R}_z^{-1}$ is the set of K triples of all component parameters. The marginal distribution of the observed data $\vec{x}$ is the weighed sum of the normal components — the GMM itself:

$$\rho(\vec{x}|\vec{\theta}) = \sum_{z=1}^{K} \rho(\vec{x}, z|\vec{\theta}) = \sum_{z=1}^{K} p_z N(\vec{x}|\vec{m}_z, \hat{R}_z). \tag{8.44}$$

For n such independent observations $\vec{x}_1, \ldots, \vec{x}_n$ (with corresponding independent hidden $z_1, \ldots, z_n$), the complete data set in the GMM is $\vec{Y} = \{\vec{y}_i\} = \{\{\vec{x}_i, z_i\}\}$ and it has, as follows from (8.43), the following distribution:

$$\rho(\vec{Y}|\vec{\theta}) = \rho(\vec{X}, \vec{Z}|\vec{\theta}) = \prod_{i=1}^{n} p_{z_i} N(\vec{x}_i|\vec{m}_{z_i}, \hat{R}_{z_i}). \tag{8.45}$$

Marginal distribution densities of observed $\vec{X} = \{\vec{x}_i\}$ and hidden $\vec{Z} = \{z_i\}$ data, according to (8.3), here are as follows:

$$\rho(\vec{X}|\vec{\theta}) = \sum_{i_1=1}^{K} \dots \sum_{i_n=1}^{K} \prod_{i=1}^{n} p_{z_i} N(\vec{x}_i|\vec{m}_{z_i}, \hat{R}_{z_i}),$$
$$\rho(\vec{Z}|\vec{\theta}) = \prod_{i=1}^{n} p_{z_i}. \tag{8.46}$$

As is defined in the previous section, each object is determined by the parameters $\vec{\mu}$ of the class it belongs to and the parameters $\vec{\nu}$ distinguishing it within the class. Thus, we can consider the GMM parameters $\vec{\mu}$ as parameters of some precedent representative for the class, which are generated at the learning stage, and $\vec{\nu}$ as the displacement parameters of this representative precedent in the Ω plane, which are calculated for the registered object at the identification stage.

Within the framework of the problem under consideration, it is advisable to confine ourselves to the class of plane affine displacements connecting the coordinates before $\{\vec{x}, \vec{z}\}$ and after $\{\vec{x}', z'\}$ transformation by the following equations:

$$\{\vec{x}, z\} \to \{\vec{x}', z'\} = \{s\hat{O}\vec{x} + \vec{t}_z, z\}, \tag{8.47}$$

where s and $\hat{O}$ are the scale factor and the orthogonal matrix $\hat{O}$ of the rotation by angle φ, the same for all components, and $\vec{t}_z$ defines the displacement vector for z-component.

Affine transformations (8.47) transform the component distributions as follows:

$$\rho'(\vec{x}', z|\vec{\theta}) = \rho(\hat{O}^T(\vec{x}' - \vec{t}_z)/s, z|\vec{\theta})/s^2. \tag{8.48}$$

For GMM, this transformation is equivalent to the transformation of distribution parameters:

$$p'_z = p_z,\ \vec{m}'_z = \vec{t}_z + s\hat{O}\vec{m}_z,\ \hat{A}'_z = \hat{O}\hat{A}_z\hat{O}^T/s^2. \tag{8.49}$$

So, each precedent class can be equivalently given by the set of $\vec{\mu} = \{(p_z, \vec{m}_z, \hat{A}_z)\}$ parameters and each object can be identified in such a class by parameters $\vec{\nu} = \{s, \hat{O}, \{\vec{t}_z\}\}$. It should be noted once

again that parameters $\vec{\nu}$ are not dependent on the class and are general for all classes.

Note that among the precedents of the given class there are always representatives with zero mean coordinates:

$$\langle \vec{x}' \rangle = \sum_{z=1}^{K} p'_z \vec{m}'_z = \sum_{z=1}^{K} p_z \vec{t}_z + s\hat{O} \sum_{z=1}^{K} p_z \vec{m}_z = \vec{0}. \tag{8.50}$$

To ensure this, it is enough to take the parameters $\vec{\nu}$ of representatives in the from $\vec{t}_z = \vec{t}$, $\vec{t} = -s\hat{O}\vec{M}$, $\vec{M} = \sum_{z=1}^{K} p_z \vec{m}_z$ for any s and $\hat{O}$. In turn, among these representatives, there is only one whose correlation matrix is diagonal and has a single trace:

$$\langle \vec{x}' \vec{x}'^T \rangle = \sum_{z=1}^{K} p'_z [\hat{R}'_z + \vec{m}'_z \vec{m}'^T_z] = \mathrm{diag}\{d_1, d_2\}, \; d_1 + d_2 = 1. \tag{8.51}$$

This representative is determined in the class by the parameters $\vec{\nu} = \{s, \hat{O}, \vec{t}_z\}$ related to the singular value decomposition of the class correlation matrix

$$\mathcal{R} = \sum_{z=1}^{K} p_z [\hat{R}_z + (\vec{m}_z - \vec{M})(\vec{m}_z - \vec{M})^T] = \hat{o}^T \mathrm{diag}\{D_1, D_2\}\hat{o} \tag{8.52}$$

using the relations $\hat{O} = \hat{o}$, $s^2 = 1/(D_1 + D_2)$.

Due to the special properties (8.50) and (8.51) of the parameters $\vec{\mu}' = \{(p'_z, \vec{m}'_z, \hat{A}'_z)\}$ of the representative considered, it is convenient instead of representing the class through the parameters of some training precedent $\vec{\mu} = \{(p_z, \vec{m}_z, \hat{A}_z)\}$ to set them through the parameters of the above representative:

$$p''_z = p'_z, \; \vec{m}''_z = \vec{t}_z + s\hat{O}\vec{m}'_z, \; \hat{A}''_z = \hat{O}\hat{A}'_z\hat{O}^T/s^2. \tag{8.53}$$

In this case, the mean and correlation functions of an arbitrary distribution in the class have the following form:

$$\begin{aligned}
&\langle \vec{x}'' \rangle = \sum_{z=1}^{K} p''_z \vec{m}''_z = \sum_{z=1}^{K} p'_z \vec{t}'_z = \vec{t}, \\
&\langle (\vec{x}'' - \vec{t})(\vec{x}'' - \vec{t})^T \rangle = s^2 \hat{O} \mathrm{diag}\{d_1, d_2\} \hat{O}^T \\
&\quad + \sum_{z=1}^{K} p_z \delta\vec{t}_z \delta\vec{t}_z^T + s\hat{O} \sum_{z=1}^{K} p_z \vec{m}'_z \delta\vec{t}_z^T + s \sum_{z=1}^{K} p_z \delta\vec{t}_z \delta\vec{m}'^T_z \hat{O}^T,
\end{aligned} \tag{8.54}$$

where $\delta\vec{t}_z = \vec{t}_z - \vec{t}$. In the case of a similar transformation of all components $\vec{t}_z = \vec{t}$ for all z, we will have $\delta\vec{t}_z = \vec{0}$ and, respectively,

$$\begin{aligned} \langle \vec{x}'' \rangle &= \vec{t}, \\ \langle (\vec{x}'' - \vec{t})(\vec{x}'' - \vec{t})^T \rangle &= \mathcal{R}'' = s^2 \hat{O} \mathrm{diag}\{d_1, d_2\} \hat{O}^T. \end{aligned} \tag{8.55}$$

8.4.2 *EM/GMM algorithm for learning procedure*

In the specified model of precedent's distribution parameters, there are transformation parameters $\vec{\nu} = \{s, \hat{O}, \{\vec{t}_z\}\}$, independent of the class, and the specific class parameters $\vec{\mu} = \{(p_z, \vec{m}_z, \hat{A}_z)\}$, which are restricted by

$$\begin{aligned} &\sum_{z=1}^{K} p_z = 1, \ \langle \vec{x} \rangle = \sum_{z=1}^{K} p_z \vec{m}_z = \vec{0}, \\ &\langle \vec{x} \cdot \vec{x}^T \rangle = \sum_{z=1}^{K} p_z [\hat{R}_z + \vec{m}_z \vec{m}_z^T] = \mathrm{diag}\{d_1, d_2\}, \end{aligned} \tag{8.56}$$

where $\hat{R}_z = \hat{A}_z^{-1}$, $d_1 + d_2 = 1$ (we redesign the parameters of the class representative in comparison with the previous section as $\{(p'_z, \vec{m}'_z, \hat{A}'_z)\} \to \{(p_z, \vec{m}_z, \hat{A}_z)\}$).

The GMM set of distribution parameters $\vec{\theta} = \{(p'_z, \vec{m}'_z, \hat{A}'_z)\}$ of any precedent in the learning class M is determined by the class $\vec{\mu} = \{(p_z, \vec{m}_z, \hat{A}_z)\}$ and transformation $\vec{\nu} = \{s, \hat{O}, \{\vec{t}_z\}\}$ parameters by the following relations ($\vec{t}_z = \vec{t}$ implied):

$$p'_z = p_z, \ \vec{m}'_z = \vec{t} + s\hat{O}\vec{m}_z, \ \hat{A}'_z = \hat{O}\, \hat{A}_z \hat{O}^T / s^2. \tag{8.57}$$

As noted above, to determine the ML parameters $\vec{\mu}_M$ from the available data $\vec{X}_M = \{\vec{x}_i\}$, the combined EM algorithm can be used with the Q-function (8.21), which in the GMM case is given by the following expression:

$$Q(\vec{X}_M; \vec{\mu}, \vec{\nu}, \vec{\mu}^{(\iota-1)}, \vec{\nu}^{(\iota-1)}) = \sum_{z_1=1}^{K} \cdots \sum_{z_n=1}^{K}$$

$$\prod_{i=1}^{n} \rho(z_i | \vec{x}_i; \vec{\theta}^{(\iota-1)}) \sum_{i=1}^{n} \ln[\rho(\vec{x}_i, z_i | \vec{\mu}, \vec{\nu})]$$

$$= \sum_{i=1}^{n}\sum_{z=1}^{K} \rho(z|\vec{x}_i;\vec{\theta}^{(\iota-1)}) \ln[\rho(\vec{x}_i, z|\vec{\mu},\vec{\nu})]$$

$$= n\left\{ -\ln[2\pi] + \sum_{z=1}^{K} \eta_z \ln[p_z] + \frac{1}{2}\sum_{z=1}^{K} \eta_z \ln[\det \hat{A}_z] \right.$$

$$- 2\ln[s] - \frac{1}{2s^2}\sum_{z=1}^{K} \eta_z Tr[\hat{O}\hat{A}_z\hat{O}^T\hat{\mathcal{R}}_z]$$

$$- \frac{1}{2}\sum_{z=1}^{K} \eta_z Tr\left[\hat{A}_z\left(\hat{O}^T\frac{(\vec{X}_z - \vec{t})}{s} - \vec{m}_z\right)\right.$$

$$\left.\left. \times \left(\hat{O}^T\frac{(\vec{X}_z - \vec{t})}{s} - \vec{m}_z\right)^T\right]\right\}, \tag{8.58}$$

where $Tr[\ldots]$ denotes the trace of the matrix, $\vec{\theta}^{(\iota-1)} = \{\vec{\mu}^{(\iota-1)}, \vec{\nu}^{(\iota-1)}\}$, and the following notations are introduced:

$$\eta_z = \frac{1}{n}\sum_{i=1}^{n} \rho(z|\vec{x}_i;\vec{\theta}^{(\iota-1)}),$$

$$\vec{X}_z = \frac{\sum_{i=1}^{n} \rho(z|\vec{x}_i;\vec{\theta}^{(\iota-1)})\vec{x}_i}{\sum_{i=1}^{n} \rho(z|\vec{x}_i;\vec{\theta}^{(\iota-1)})}, \tag{8.59}$$

$$\hat{\mathcal{R}}_z = \frac{\sum_{i=1}^{n} \rho(z|\vec{x}_i;\vec{\theta}^{(\iota-1)})(\vec{x}_i - \vec{X}_z)(\vec{x}_i - \vec{X}_z)^T}{\sum_{i=1}^{n} \rho(z|\vec{x}_i;\vec{\theta}^{(\iota-1)})},$$

where implied

$$\rho(z|\vec{x}_i;\vec{\theta}^{(\iota-1)})$$

$$= \frac{p_z^{(\iota-1)}\sqrt{\det \hat{A}_z^{(\iota-1)}}\exp\{-\vec{\xi}_{i,z}^T\hat{A}_z^{(\iota-1)}\vec{\xi}_{i,z}\}}{\sum_{j=1}^{K} p_j^{(\iota-1)}\sqrt{\det \hat{A}_j^{(\iota-1)}}\exp\{-\vec{\xi}_{i,j}^T\hat{A}_j^{(\iota-1)}\vec{\xi}_{i,j}\}},$$

$$\vec{\xi}_{i,z} = \frac{1}{s^{(\iota-1)}}\hat{O}^{(\iota-1)T}(\vec{x}_i - \vec{t}^{(\iota-1)}) - \vec{m}_z^{(\iota-1)}. \tag{8.60}$$

In view of the restrictions (8.56) on $\vec{\mu} = \{(p_z, \vec{m}_z, \hat{A}_z)\}$, while maximizing $Q(\vec{X}_M;\vec{\mu},\vec{\nu},\vec{\mu}^{(\iota-1)},\vec{\nu}^{(\iota-1)})$ (8.58), it is advisable to use

the method of Lagrange multipliers expanding the Q-function to the Lagrangian:

$$\mathcal{L}(\vec{X}_M;\vec{\mu},Q(\vec{X}_M;\vec{\mu},\vec{\nu},\vec{\theta}^{(\iota-1)})) = Q(\vec{X}_M;\vec{\mu},Q(\vec{X}_M;\vec{\mu},\vec{\nu},\vec{\theta}^{(\iota-1)})) - \lambda\sum_{z=1}^{K} p_z - \vec{\zeta}\sum_{z=1}^{K} p_z\vec{m}_z - \xi Tr\left(\sum_{z=1}^{K} p_z[\hat{R}_z + \vec{m}_z\vec{m}_z^T]\right) - \varsigma Tr\left(\begin{pmatrix}0 & 1\\ 1 & 0\end{pmatrix}\sum_{z=1}^{K} p_z[\hat{R}_z + \vec{m}_z\vec{m}_z^T]\right), \tag{8.61}$$

where λ, $\vec{\zeta}$, ξ, ς are Lagrange multipliers, and for $\mathcal{L}$, perform the unconditional maximization.

Without going into the details of the calculations, let us present only the results. The optimal values for the parameters $\{\vec{\mu}_M,\vec{\nu}\}$ are as follows:

$$p_z = \eta_z,\ \vec{m}_z = \frac{1}{s}\hat{O}^T(\vec{X}_z - \vec{t}),\ \hat{A}_z = s^2\hat{O}^T\hat{\mathcal{R}}_z^{-1}\hat{O}, \tag{8.62}$$

$$\vec{t} = \vec{X},\ s = \sqrt{\mathcal{R}_{11}+\mathcal{R}_{22}},$$

$$\hat{O} = \begin{pmatrix}\cos\varphi & -\sin\varphi\\ \sin\varphi & \cos\varphi\end{pmatrix},\ \varphi = -\frac{1}{2}\mathrm{arctg}\frac{2\mathcal{R}_{12}}{\mathcal{R}_{22}-\mathcal{R}_{11}}, \tag{8.63}$$

where are introduced the following notations:

$$\vec{X} = \frac{\sum_{i=1}^{n}\vec{x}_i}{n},\ \hat{\mathcal{R}} = \frac{\sum_{i=1}^{n}(\vec{x}_i-\vec{X})(\vec{x}_i-\vec{X})^T}{n}. \tag{8.64}$$

Note that parameters $\vec{\nu} = \{s,\hat{O},\vec{t}\}$ are dependent only on the data $\vec{X}_M = \{\vec{x}_i\}$ and are not dependent on the previous estimates $\vec{\theta}^{(\iota-1)} = \{\vec{\mu}^{(\iota-1)},\vec{\nu}^{(\iota-1)}\}$. For this reason, they can be computed *prior* to the start of the learning procedure and not recomputed at the iterations.

In the compressed form, the results of the analysis can be compiled into the following EM-learning algorithm:

Input data: Observed data $\vec{X}_M = \{\vec{x}_i\}$, $i = 1,\ldots,n$, corresponding to the training precedent of class M; the number K of GMM components, threshold $\varepsilon > 0$ for the stopping algorithm criterion.

Initialization: Calculation of sampling mean vector $\vec{X}$ and sampling correlation matrix $\hat{\mathcal{R}}$ (8.64), and, with their help, calculation of the parameters $\vec{\nu} = \{s, \hat{O}, \vec{t}\}$ (8.63).

Initialization of sampling means of components $\{\vec{X}_z\}$ (8.64), using some random subsamples $\{\vec{x}_z\}$ from $\vec{X}_M = \{\vec{x}_i\}$, $z = 1, \ldots, K$: $\vec{X}_z = \vec{x}_z$.

Calculation of sampling correlation matrixes of components $\{\hat{\mathcal{R}}_z\}$ (8.64), assuming the same distribution $\{p_z\}$ for all component indicators $\{z_i\}$ (for example, uniform):

$$\hat{\mathcal{R}}_z = \frac{\sum_{i=1}^{n} (\vec{x}_i - \vec{X}_z)(\vec{x}_i - \vec{X}_z)^T}{n}. \tag{8.65}$$

Setting the initial values of class parameters $\vec{\mu}^{(0)}$:

$$p_z^{(0)} = p_z,\ \vec{m}_z^{(0)} = \frac{1}{s}\hat{O}^T(\vec{X}_z - \vec{t}),\ \hat{A}_z^{(0)} = s^2\hat{O}^T\hat{\mathcal{R}}_z^{-1}\hat{O}. \tag{8.66}$$

Based on the initial values of all the above parameters, the calculation of the initial *posteriori* distributions of component indicators $\{z_i\}$:

$$\begin{aligned}
\rho_z^{(0)}(\vec{x}_i) &= \rho(z|\vec{x}_i; \vec{\mu}^{(0)}, \vec{\nu}) \\
&= \frac{p_z^{(0)}\sqrt{\det \hat{A}_z^{(0)}}\,\exp\{-\vec{\xi}_{i,z}^T \hat{A}_z^{(0)} \vec{\xi}_{i,z}\}}{\sum_{j=1}^{K} p_j^{(0)}\sqrt{\det \hat{A}_j^{(0)}}\,\exp\{-\vec{\xi}_{i,j}^T \hat{A}_j^{(0)} \vec{\xi}_{i,j}\}}, \\
\vec{\xi}_{i,z} &= \frac{1}{s}\hat{O}^T(\vec{x}_i - \vec{t}) - \vec{m}_z^{(0)}.
\end{aligned} \tag{8.67}$$

Initialization of the iteration counter $\iota = 1$.

E-step: Using *posteriori* distribution of the indicators $\{\rho_z^{(\iota-1)}(\vec{x}_i)\}$, available at iteration (ι), the sampling parameters are calculated:

$$\begin{aligned}
\eta_z &= \frac{1}{n}\sum_{i=1}^{n} \rho_z^{(\iota-1)}(\vec{x}_i),\ \vec{X}_z = \frac{\sum_{i=1}^{n} \rho_z^{(\iota-1)}(\vec{x}_i) \cdot \vec{x}_i}{n\eta_z}, \\
\hat{\mathcal{R}}_z &= \frac{\sum_{i=1}^{n} \rho_z^{(\iota-1)}(\vec{x}_i) \cdot (\vec{x}_i - \vec{X}_z)(\vec{x}_i - \vec{X}_z)^T}{n\eta_z}.
\end{aligned} \tag{8.68}$$

M-step: Using the sampling parameters found, the new values $\vec{\mu}^{(\iota)}$ are calculated:

$$p_z^{(\iota)} = \eta_z;\ \vec{m}_z^{(\iota)} = \frac{1}{s}\hat{O}^T(\vec{X}_z - \vec{t});\ \hat{A}_z^{(\iota)} = s^2\hat{O}^T\hat{\mathcal{R}}_z^{-1}\hat{O}. \tag{8.69}$$

For all component indicators $\{z_i\}$, *posteriori* distributions are recalculated:

$$\rho_z^{(\iota)}(\vec{x}_i) = \rho(z|\vec{x}_i;\vec{\mu}^{(\iota)},\vec{\nu}) = \frac{p_z^{(\iota)}\sqrt{\det \hat{A}_z^{(\iota)}}\exp\{-\vec{\xi}_{i,z}^T\hat{A}_z^{(\iota)}\vec{\xi}_{i,z}\}}{\sum_{j=1}^K p_j^{(\iota)}\sqrt{\det \hat{A}_j^{(0)}}\exp\{-\vec{\xi}_{i,j}^T\hat{A}_j^{(\iota)}\vec{\xi}_{i,j}\}},$$

$$\vec{\xi}_{i,z} = \frac{1}{s}\hat{O}^T(\vec{x}_i - \vec{t}) - \vec{m}_z^{(0)}. \tag{8.70}$$

Based on the new $\{\rho_z^{(\iota)}(\vec{x}_i)\}$ and previous $\{\rho_z^{(\iota-1)}(\vec{x}_i)\}$ *posteriori* distributions, an approximation error is constructed (Kullback–Leibler divergence):

$$\delta^{(\iota-1)}(\vec{X}_M;\vec{\mu}^{(\iota)},\vec{\nu}) = D_{KL}(\vec{X}_M;\vec{\mu}^{(\iota)},\vec{\nu},\vec{\mu}^{(\iota-1)},\vec{\nu})$$

$$= -\sum_{i=1}^n\sum_{z=1}^K \rho_z^{(\iota-1)}(\vec{x}_i)\ln\left[\frac{\rho_z^{(\iota)}(\vec{x}_i)}{\rho_z^{(\iota-1)}(\vec{x}_i)}\right]. \tag{8.71}$$

If $|\delta^{(\iota-1)}(\vec{X}_M;\vec{\theta}^{(j)})| > \varepsilon$, then iteration counter ι is incremented and proceeds to step E to perform the next iteration of the algorithm. Otherwise, the algorithm terminates and $\vec{\mu}_M = \vec{\mu}^{(\iota)} = \{(p_z^{(\iota)},\vec{m}_z^{(\iota)},\hat{A}_z^{(\iota)})\}$ is accepted as class M parameters.

8.4.3 *EM/GMM algorithm for identification procedure*

As in the previous section, the distribution parameters are transformation parameters $\vec{\nu} = \{s,\hat{O},\{\vec{t}_z\}\}$, independent of the class, and the specific class parameters $\vec{\mu}_M = \{(p_z^M,\vec{m}_z^M,\hat{A}_z^M)\}$ with restrictions (8.56).

The GMM set of distribution parameters $\vec{\theta} = \{(p'_z,\vec{m}'_z,\hat{A}'_z)\}$ of some class M precedent is determined by the parameters $\{\vec{\mu}_M,\vec{\nu}\}$

and by the following relations:

$$p'_z = p_z^M,\ \vec{m}'_z = \vec{t}_z + s\hat{O}\vec{m}_z^M,\ \hat{A}'_z = \hat{O}\,\hat{A}_z^M\hat{O}^T/s^2. \tag{8.72}$$

As noted above, to determine the ML label M_{ML} of the identified object class from the available data $\vec{X} = \{\vec{x}_i\}$, the EM algorithm can be used with the Q-function (8.33), which in the case of GMM is given by the following expression:

$$\begin{aligned}
&Q(\vec{X};\vec{\mu}_M,\vec{\mu}_N,\vec{\nu}_N)\\
&\quad=\sum_{z_1=1}^{K}\cdots\sum_{z_{n=1}}^{K}\prod_{i=1}^{n}\rho(z_i|\vec{x}_i;\vec{\theta}_N)\sum_{i=1}^{n}\ln[\rho(\vec{x}_i,z_i|\vec{\mu}_M,\vec{\nu}_N)]\\
&\quad=\sum_{i=1}^{n}\sum_{z=1}^{K}\rho(z|\vec{x}_i;\vec{\theta}_N)\ln[\rho(\vec{x}_i,z|\vec{\mu}_M,\vec{\nu}_N)]\\
&\quad=n\left\{-\ln[2\pi]+\sum_{z=1}^{K}\eta_z\ln[p_z^M]+\frac{1}{2}\sum_{z=1}^{K}\eta_z\ln[\det\hat{A}_z^M]\right.\\
&\qquad-2\ln[s^N]-\frac{1}{2s^{N^2}}\sum_{z=1}^{K}\eta_z Tr[\hat{O}^N\hat{A}_z^M\hat{O}^{NT}\hat{\mathcal{R}}_z]\\
&\qquad-\frac{1}{2}\sum_{z=1}^{K}\eta_z Tr\left[\hat{A}_z^M\left(\hat{O}^{NT}\frac{(\vec{X}_z-\vec{t}_z^N)}{s^N}-\vec{m}_z^M\right)\right.\\
&\qquad\left.\left.\times\left(\hat{O}^{NT}\frac{(\vec{X}_z-\vec{t}_z^N)}{s^N}-\vec{m}_z^M\right)^T\right]\right\},
\end{aligned} \tag{8.73}$$

where $Tr[\ldots]$ denotes the trace of the matrix, $\vec{\theta}_N = \{\vec{\mu}_N,\vec{\nu}_N\}$, and the following notations are introduced:

$$\begin{aligned}
\eta_z &= \frac{1}{n}\sum_{i=1}^{n}\rho(z|\vec{x}_i;\vec{\theta}_N),\ \vec{X}_z = \frac{\sum_{i=1}^{n}\rho(z|\vec{x}_i;\vec{\theta}_N)\vec{x}_i}{\sum_{i=1}^{n}\rho(z|\vec{x}_i;\vec{\theta}_N)},\\
\hat{\mathcal{R}}_z &= \frac{\sum_{i=1}^{n}\rho(z|\vec{x}_i;\vec{\theta}_N)(\vec{x}_i-\vec{X}_z)(\vec{x}_i-\vec{X}_z)^T}{\sum_{i=1}^{n}\rho(z|\vec{x}_i;\vec{\theta}_N)}
\end{aligned} \tag{8.74}$$

and implied

$$\rho(z|\vec{x}_i;\vec{\theta}_N) = \frac{p_z^N \sqrt{\det \hat{A}_z^N}\exp\{-\vec{\xi}_{i,z}^T \hat{A}_z^N \vec{\xi}_{i,z}\}}{\sum_{j=1}^K p_j^N \sqrt{\det \hat{A}_j^N}\exp\{-\vec{\xi}_{i,j}^T \hat{A}_j^N \vec{\xi}_{i,j}\}}, \qquad (8.75)$$

$$\vec{\xi}_{i,z} = \hat{O}^{NT}\frac{(\vec{x}_i - \vec{t}_z^N)}{s^N} - \vec{m}_z^N.$$

Unlike the learning problem presented in the previous section, the EM algorithm for the identification problem cannot jointly calculate all parameters, so the parameters $\vec{\nu}_N = \{s^N, \hat{O}^N, \{\vec{t}_z^N\}\}$ should be calculated at every E-step within the internal E′ M′ algorithm with q-function (8.36), which in the GMM case is given by the following expression:

$$q(\vec{X};\vec{\nu},\vec{\mu}_N,\vec{\nu}^{(\kappa-1)}) = \sum_{z_1=1}^K \cdots \sum_{z_{n=1}}^K \prod_{i=1}^n \rho(z_i|\vec{x}_i;\vec{\mu}_N,\vec{\nu}^{(\kappa-1)})$$

$$\prod_{i=1}^n \rho(z_i|\vec{x}_i;\vec{\mu}_N,\vec{\nu}^{(\kappa-1)}) \sum_{i=1}^n \ln[\rho(\vec{x}_i,z_i|\vec{\mu}_N,\vec{\nu})]$$

$$= \sum_{i=1}^n \sum_{z=1}^K \rho(z|\vec{x}_i;\vec{\mu}_N,\vec{\nu}^{(\kappa-1)})\ln[\rho(\vec{x}_i,z|\vec{\mu}_N,\vec{\nu})]$$

$$= n\left\{ -\ln[2\pi] + \sum_{z=1}^K \eta_z \ln[p_z^N] + \frac{1}{2}\sum_{z=1}^K \eta_z \ln[\det \hat{A}_z^N] \right.$$

$$- 2\ln[s] - \frac{1}{2s^2}\sum_{z=1}^K \eta_z Tr[\hat{O}\hat{A}_z^N\hat{O}^T\hat{\mathcal{R}}_z]$$

$$- \frac{1}{2}\sum_{z=1}^K \eta_z Tr\left[\hat{A}_z^N\left(\hat{O}^T\frac{(\vec{X}_z - \vec{t}_z)}{s} - \vec{m}_z^N\right)\right.$$

$$\left.\left. \times \left(\hat{O}^T\frac{(\vec{X}_z - \vec{t}_z)}{s} - \vec{m}_z^N\right)^T\right]\right\}, \qquad (8.76)$$

where

$$\eta_z = \frac{1}{n}\sum_{i=1}^{n} \rho(z|\vec{x}_i;\vec{\mu}_N,\vec{\nu}^{(\kappa-1)}),\ \vec{X}_z = \frac{\sum_{i=1}^{n}\rho(z|\vec{x}_i;\vec{\mu}_N,\vec{\nu}^{(\kappa-1)})\vec{x}_i}{\sum_{i=1}^{n}\rho(z|\vec{x}_i;\vec{\mu}_N,\vec{\nu}^{(\kappa-1)})},$$

$$\hat{R}_z = \frac{\sum_{i=1}^{n}\rho(z|\vec{x}_i;\vec{\mu}_N,\vec{\nu}^{(*)})(\vec{x}_i-\vec{X}_z)(\vec{x}_i-\vec{X}_z)^T}{\sum_{i=1}^{n}\rho(z|\vec{x}_i;\vec{\mu}_N,\vec{\nu}^{(\kappa-1)})}, \tag{8.77}$$

and implied

$$\rho(z|\vec{x}_i;\vec{\mu}_N,\vec{\nu}^{(\kappa-1)}) = \frac{p_z^N\sqrt{\det\hat{A}_z^N}\exp\{-\vec{\xi}_{i,z}^T\hat{A}_z^N\vec{\xi}_{i,z}\}}{\sum_{j=1}^{K}p_j^N\sqrt{\det\hat{A}_j^N}\exp\{-\vec{\xi}_{i,j}^T\hat{A}_j^N\vec{\xi}_{i,j}\}},$$

$$\vec{\xi}_{i,z} = \hat{O}^{(\kappa-1)T}\frac{(\vec{x}_i-\vec{t}_z^{(\kappa-1)})}{s^{(\kappa-1)}} - \vec{m}_z^N. \tag{8.78}$$

Since the parameters $\vec{\nu} = \{s,\hat{O},\{\vec{t}_z\}$ are not bound by any restrictions, unconditional maximization of the q-function (8.76) is admissible here. Without going into the details of the calculations, let us present only the results. The optimal values for the parameters $\vec{\nu}$ are as follows:

$$\begin{aligned}
\vec{t}_z &= \vec{X}_z - s\hat{O}\vec{m}_z^N,\\
\hat{O} &= \begin{pmatrix}\cos\varphi & -\sin\varphi\\ \sin\varphi & \cos\varphi\end{pmatrix},\\
\varphi &= \frac{1}{2}\mathrm{arctg}\frac{\sum_{z=1}^{K}\eta_z dAz\cdot d\mathcal{R}z\sin 2(\psi_z-\chi_z)}{\sum_{z=1}^{K}\eta_z dAz\cdot d\mathcal{R}z\cos 2(\psi_z-\chi_z)},\\
s^2 &= \frac{1}{4}\sum_{z=1}^{K}\eta_z tAz\cdot t\mathcal{R}z\\
&\quad + \frac{1}{4}\cos 2\varphi\sum_{z=1}^{K}\eta_z dAz\cdot d\mathcal{R}z\cos 2(\psi_z-\chi_z)\\
&\quad + \frac{1}{4}\sin 2\varphi\sum_{z=1}^{K}\eta_z dAz\cdot d\mathcal{R}z\sin 2(\psi_z-\chi_z),
\end{aligned} \tag{8.79}$$

where parameters ψ_z, tAz, dAz are calculated by the component matrixes of the $\hat{A}_z^N$ of the class parameters $\vec{\mu}_N$:

$$\psi_z = \frac{1}{2}\mathrm{arctg}\frac{2a_{z12}}{a_{z22}-a_{z11}}, \quad tAz = a_{z11}+a_{z22},$$
$$dAz = \mathrm{sign}(a_{z22}-a_{z11})\sqrt{(a_{z11}+a_{z22})^2 - 4(a_{z11}a_{z22}-a_{z12}^2)}, \tag{8.80}$$

and the parameters χ_z, $t\mathcal{R}z$, $d\mathcal{R}z$ are calculated by the corresponding sample component matrixes $\hat{\mathcal{R}}_z$ (8.77):

$$\chi_z = \frac{1}{2}\mathrm{arctg}\frac{2r_{z12}}{r_{z22}-a_{z11}}, \quad t\mathcal{R}z = r_{z11}+r_{z22},$$
$$d\mathcal{R}z = \mathrm{sign}(r_{z22}-r_{z11})\sqrt{(r_{z11}+r_{z22})^2 - 4(r_{z11}r_{z22}-r_{z12}^2)}. \tag{8.81}$$

In a compressed form, the results of the above analysis can be compiled into the following EM-identification algorithm:

Input data: Observed data $\vec{X} = \{\vec{x}_i\}$, $i = 1,\ldots,n)\}$, database of classes and their descriptions $\vec{\mu}_M$, $M \in \{M\}$ in the form of the sets of triples $\vec{\mu}_M = \{(p_z^M, m_z^M, \hat{A}_z^M)\}$, threshold $\varepsilon > 0$ for the stopping criterion of the internal E′M′ .

Initialization: Calculation (as in the learning procedure) of sample mean $\vec{X}$ and sample correlation matrix $\hat{\mathcal{R}}$ (8.69), and, with their help, calculation of the initial estimations $\vec{\nu}^0 = \{s^0, \hat{O}^0, \vec{t}^{\,0}\}$ (analogous to (8.63)):

$$\vec{t}^{\,0} = \vec{X}, \; s^0 = \sqrt{\mathcal{R}_{11}+\mathcal{R}_{22}},$$
$$\hat{O}^0 = \begin{pmatrix} \cos\varphi & -\sin\varphi \\ \sin\varphi & \cos\varphi \end{pmatrix}, \quad \varphi = -\frac{1}{2}\mathrm{arctg}\frac{2\mathcal{R}_{12}}{\mathcal{R}_{22}-\mathcal{R}_{11}}. \tag{8.82}$$

Selection of some initial class 0 with parameters $\vec{\mu}_0 = \{(p_z^0, \vec{m}_z^0, \hat{A}_z^0)\}$.

Initialization of the iteration counter $\iota = 1$.

E-step: In order to find $\vec{\nu}_N$ parameters, which are optimal for the class N, available at iteration (ι), for all $\hat{A}_z^N$ from the set $\vec{\mu}_N = \{(p_z^N, \vec{m}_z^N, \hat{A}_z^N)\}$ parameters ψ_z, tAz, dAz (8.85) are calculated.

After that begins internal E′ M′:

Initialization: Initial values of $\vec{\nu}^{(0)} = \{s^{(0)}, \hat{O}^{(0)}, \{\vec{t}_z^{(0)}\}\}$ are set by means of estimates $\vec{\nu}^0$ (8.82):

$$s^{(0)} = s^0,\ \hat{O}^{(0)} = \hat{O}^0,\ \vec{t}_z^{(0)} = \vec{t}^0,\ z = 1, \ldots, K. \tag{8.83}$$

Based on the initial values $\vec{\nu}^{(0)}$, initial *posteriori* distributions of component indicators $\{z_i\}$ are calculated:

$$\rho_z^{(0)}(\vec{x}_i) = \rho(z|\vec{x}_i; \vec{\mu}_N, \vec{\nu}^{(0)}) = \frac{p_z^N \sqrt{\det \hat{A}_z^N}\, \exp\{-\vec{\xi}_{i,z}^T \hat{A}_z^N \vec{\xi}_{i,z}\}}{\sum_{j=1}^K p_j^N \sqrt{\det \hat{A}_j^N}\, \exp\{-\vec{\xi}_{i,j}^T \hat{A}_j^N \vec{\xi}_{i,j}\}},$$

$$\vec{\xi}_{i,z} = \hat{O}^{(0)T} \frac{(\vec{x}_i - \vec{t}_z^{(0)})}{s^{(0)}} - \vec{m}_z^N. \tag{8.84}$$

Initialization of the iteration counter $\kappa = 1$.

E′-step: Using *posteriori* distribution of the indicators $\{\rho_z^{(\kappa-1)}(\vec{x}_i)\}$, available at iteration (ξ), sample parameters are calculated:

$$\begin{aligned} \eta_z &= \frac{1}{n} \sum_{i=1}^n \rho_z^{(\kappa-1)}(\vec{x}_i),\ \vec{X}_z = \frac{\sum_{i=1}^n \rho_z^{(\kappa-1)}(\vec{x}_i) \cdot \vec{x}_i}{n\eta_z}, \\ \hat{\mathcal{R}}_z &= \frac{\sum_{i=1}^n \rho_z^{(\kappa-1)}(\vec{x}_i) \cdot (\vec{x}_i - \vec{X}_z)(\vec{x}_i - \vec{X}_z)^T}{n\eta_z}, \end{aligned} \tag{8.85}$$

and for all $\hat{\mathcal{R}}_z$ parameters ψ_z, tAz, dAz (8.86) are calculated.

M′-step: Using the parameters found, the new values $\vec{\nu}^{(\kappa)}$ are calculated:

$$\varphi = \frac{1}{2} \mathrm{arctg} \frac{\sum_{z=1}^K \eta_z dAz \cdot d\mathcal{R}z \sin 2(\psi_z - \chi_z)}{\sum_{z=1}^K \eta_z dAz \cdot d\mathcal{R}z \cos 2(\psi_z - \chi_z)},$$

$$\begin{aligned} s^{(\kappa)^2} &= \frac{1}{4} \sum_{z=1}^K \eta_z tAz \cdot t\mathcal{R}z \\ &\quad + \frac{1}{4} \cos 2\varphi \sum_{z=1}^K \eta_z dAz \cdot d\mathcal{R}z \cos 2(\psi_z - \chi_z) \end{aligned}$$

$$+\frac{1}{4}\sin 2\varphi\sum_{z=1}^{K}\eta_z dAz\cdot d\mathcal{R}z\sin 2(\psi_z-\chi_z),$$

$$\hat{O}^{(\kappa)}=\begin{pmatrix}\cos\varphi & -\sin\varphi\\ \sin\varphi & \cos\varphi\end{pmatrix},\ \vec{t}_z^{(\kappa)}=\vec{X}_z-s^{(\kappa)}\hat{O}^{(\kappa)}\vec{m}_z^N. \quad (8.86)$$

For all component indicators $\{z_i\}$, *posteriori* distributions are recalculated:

$$\rho_z^{(\kappa)}(\vec{x}_i)=\rho(z|\vec{x}_i;\vec{\mu}_N,\vec{\nu}^{(\kappa)})=\frac{p_z^N\sqrt{\det\hat{A}_z^N}\exp\{-\vec{\xi}_{i,z}^T\hat{A}_z^N\vec{\xi}_{i,z}\}}{\sum_{j=1}^K p_j^N\sqrt{\det\hat{A}_j^N}\exp\{-\vec{\xi}_{i,j}^T\hat{A}_j^N\vec{\xi}_{i,j}\}},$$

$$\vec{\xi}_{i,z}=\hat{O}^{(\kappa)T}\frac{(\vec{x}_i-\vec{t}_z^{(\kappa)})}{s^{(\kappa)}}-\vec{m}_z^N. \quad (8.87)$$

Based on the new $\{\rho_z^{(\kappa)}(\vec{x}_i)\}$ and previous $\{\rho_z^{(\kappa-1)}(\vec{x}_i)\}$ *posteriori* distributions, an approximation error is constructed (Kullback–Leibler divergence):

$$\delta^{(\kappa-1)}(\vec{X};\vec{\nu}^{(\kappa)},\vec{\mu}_N)=D_{KL}(;\vec{X};\vec{\nu}^{(\kappa)},\vec{\mu}_N,\vec{\nu}^{(\kappa-1)})$$
$$=-\sum_{i=1}^{n}\sum_{z=1}^{K}\rho_z^{(\kappa-1)}(\vec{x}_i)\ln\left[\frac{\rho_z^{(\kappa)}(\vec{x}_i)}{\rho_z^{(\kappa-1)}(\vec{x}_i)}\right]. \quad (8.88)$$

If $|\delta^{(\kappa-1)}(\vec{X};\vec{\nu}^{(\kappa)},\vec{\mu}_N)|>\varepsilon$, then iteration counter κ is incremented and proceeds to step E′ to perform the next iteration of the internal algorithm. Otherwise, the internal E′ M′ terminates and $\vec{\nu}_N=\vec{\nu}^{(\kappa)}$ is accepted.

Based on the distributions $\rho_z^{(\kappa)}(\vec{x}_i)$ (8.88) obtained in the frames of the internal E′ M′ algorithm, the sampling parameters for the main algorithm are calculated:

$$\eta_z=\frac{1}{n}\sum_{i=1}^{n}\rho(z|\vec{x}_i;\vec{\mu}_N,\vec{\nu}^{(\kappa)}),$$

$$\vec{X}_z=\frac{\sum_{i=1}^n\rho(z|\vec{x}_i;\vec{\mu}_N,\vec{\nu}^{(\kappa)})\vec{x}_i}{\sum_{i=1}^n\rho(z|\vec{x}_i;\vec{\mu}_N,\vec{\nu}^{(\kappa)})}, \quad (8.89)$$

$$\hat{\mathcal{R}}_z=\frac{\sum_{i=1}^n\rho(z|\vec{x}_i;\vec{\mu}_N,\vec{\nu}^{(\kappa)})(\vec{x}_i-\vec{X}_z)(\vec{x}_i-\vec{X}_z)^T}{\sum_{i=1}^n\rho(z|\vec{x}_i;\vec{\mu}_N,\vec{\nu}^{(\kappa)})},$$

and further, for all available classes, the values of Q-functions (8.73) are calculated.

M-step: A new class label $N^{(*)}$ is selected, based on (8.32):

$$N^{(*)} = arg \max_{M}(Q(\vec{X}; \vec{\mu}_M, \vec{\mu}_N, \vec{\nu}_N)). \tag{8.90}$$

If $N^{(*)} = N$, then $M^{(*)} = N^{(*)}$ is accepted and the algorithm terminates, if not, then $N = N^{(*)}$ is accepted as the current refinement of the $M^{(*)}$ and the algorithm proceeds to **E-step**.

8.4.4 *Short illustrative materials*

In this section, for the purpose of illustrating the proposed identification approach, the results of a computational experiment for a small, artificially formed database are presented. Descriptions of precedents in this database were obtained in processing registration data of the objects with the simple structure. The latter are chosen as binary images from the well-known base "MPEG-7 Core Experiment CE-Shape-1" [9]. This image base is usually used to compare the shape recognition algorithms and includes 70 categories of objects with 20 images in each category.

From the MPEG-7 database were arbitrarily chosen the images of five different categories: "apple-10", "butterfly-15", "cup-19", "flatfish-10", and "tree-14" as precedents. For unification, they were reduced to a size of 256×256 pixels and centered in the frames. Since the form of the intensity is simple, uniform over the objects, the simulation of registration data of precedents was trivial, 1000 counts of the first uniformly distributed across its shape. Some examples of the precedent objects and corresponding registration data (1000 counts) are presented in Figure 8.2.

To form a description of the selected precedents in the form of Gaussian mixtures with $K = 10$ components (for all precedents), the EM algorithm discussed in equations (8.65)–(8.70) was applied. Figure 8.3 shows the results of the initialization and application of the EM algorithm for the formation of the "tree-14" precedent description (see Figure 8.2). The components of the initial and resulting description are represented by centers and ellipses of constant level $\sim e^{-2}$ from the absolute maximum of the entire Gaussian mixture.

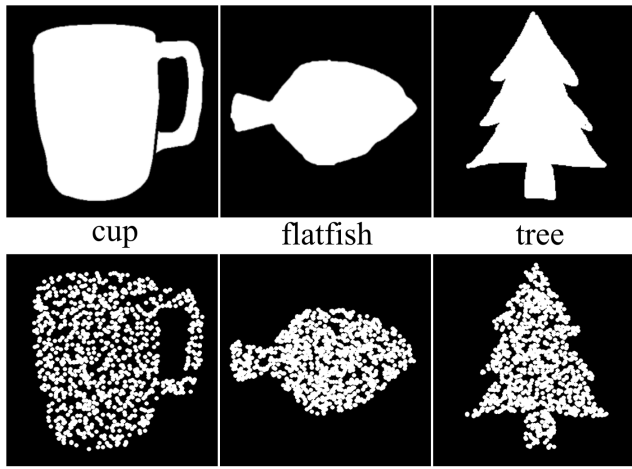


Figure 8.2. Precedents from the "MPEG-7" database [9] represented by 256×256 binary images (top) and their registration model data as 1000 counts, uniformly distributed over their shapes (bottom).

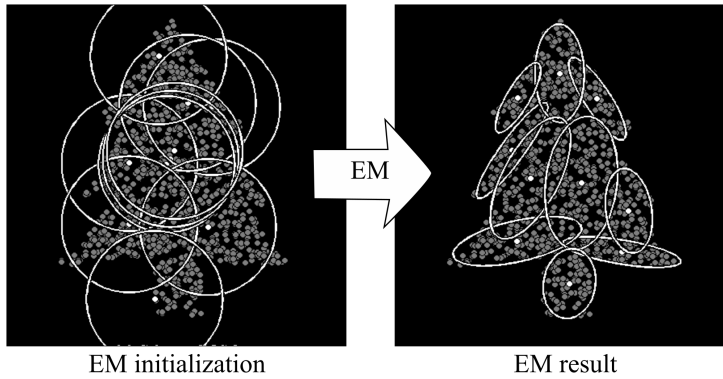


Figure 8.3. The results of the initialization (on the left) and the application (on the right) of the EM algorithm for the formation of the "tree-14" precedent description. The components of the corresponding mixtures are represented by their centers (points) and lines (ellipses) of a constant level, approximately 0.1 times the maximum value of the entire distribution.

As the test objects, the images "apple-20", "butterfly-13", "cup-7", "flatfish-3", and "tree-9" were selected from the same five categories as the precedents. All of them were reduced to the same

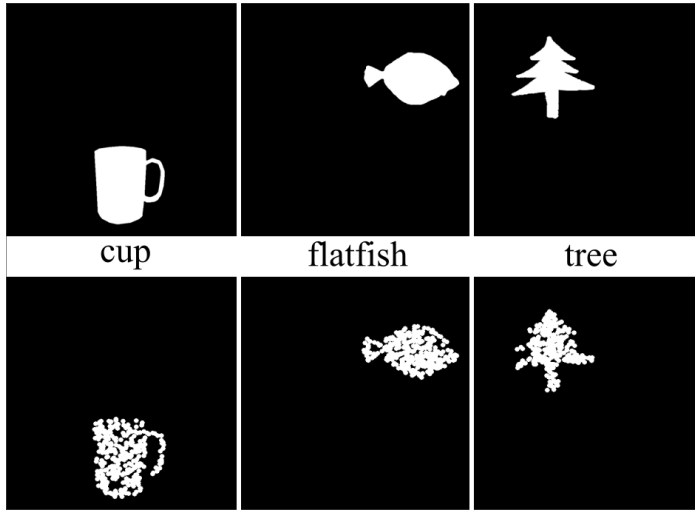


Figure 8.4. Test objects from the database "MPEG-7" [9] are selected from the same categories as the precedents in Figure 8.2 but shifted and reduced randomly (top) and model data of their registration in the form of uniformly distributed 300 counts (bottom).

frame size of 512×512 points and randomly disposed from the center of the frame and reduced in scale. The simulation of registration data was a selection of 300 generated random points, uniformly distributed within the shapes of the objects. The test objects and the corresponding registration data (300 samples) are presented in Figure 8.4.

For each of the test objects, an array of log-likelihood values was calculated for each precedent. Essentially, as noted above, the log-likelihood values based on the proposed procedure are only approximations. Nevertheless, assuming that $\{\vec{x}_i\}$ are the result of registering the counts of the M–precedent after affine transformation with ML parameters $\vec{\nu}^{(*)}$, the new (for each precedent) descriptions $\vec{\mu}_M$ for $M =$ "apple-10", "butterfly-15", "cup-19", "Flatfish-10", and "tree-14" were calculated and the values of the corresponding log-likelihoods for the test object "tree-9" were refined. The results are presented in Figure 8.5.

The full results of the calculation based on the proposed identification procedure, including other test objects "apple-20", "butterfly-13", "cup-7", and "flatfish-3" (see Figure 8.3), are shown in Table 8.1.

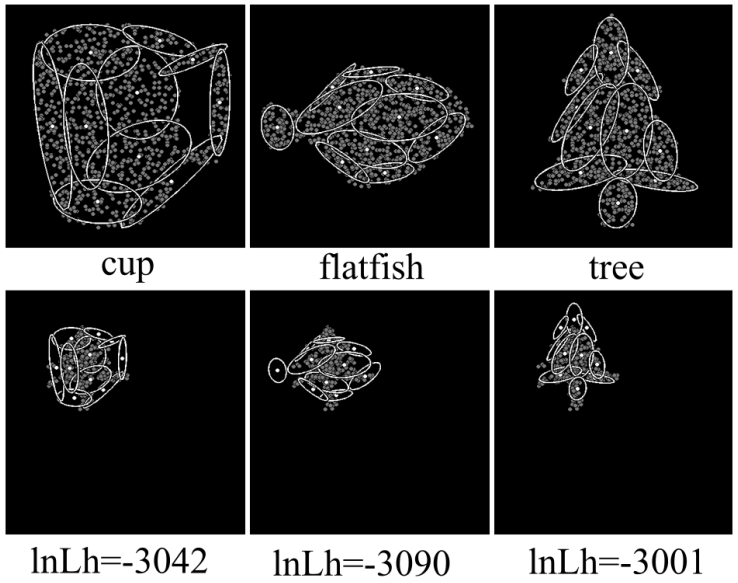


Figure 8.5. Descriptions of precedents from the database (top) and their recalculated versions using the found maximum-likelihood parameters $\vec{\nu}^{(*)}$ and descriptions (bottom) for the registration data of the test object "tree-9" (see Figure 8.4). In the following for each precedent, the logarithm of the likelihoods denoted as "lnLh" is presented.

Table 8.1. Values of the log-likelihood for test objects of different categories.

Test objects	**Precedents**		
↓	cup-19	flatfish-10	tree-14
cup-7	–3029	–3129	–3174
flatfish-3	–3175	–2949	–3193
tree-9	–3042	–3090	–3001

Note that the identification based on the criterion (8.90) of the target precedent (category) in all cases is correct. Of course, for an objective, statistically reliable assessment of the approach proposed, it is necessary to test it on much larger databases, however, the results obtained already suggest optimistic predictions regarding its potential.

8.5 Conclusions

This chapter proposes a way of formalizing the process of radiation registration by photon-counting sensors based on the PPP model [12]. It is shown that this approach is both the most adequate from the physical (quantum) point of view [6] and extremely fruitful for problems of statistical inference [8]. On this basis, using the principles of machine learning, we succeeded in developing an effective approach to the procedure synthesis both for the learning step and for the step of identification (recognition) of objects by their shape. Algorithmic realization of procedures belongs to a well-proven family of EM algorithms. The numerical simulation showed that the synthesized identification procedure has a high convergence rate. For example, in the case of describing precedents by GMM with $K \sim 10$ components, the recurrent calculations converge in less than 10 iterations.

References

[1] Antsiperov, V. (2018) Precedent–based low count rate image intensity estimation using maximum likelihood distribution descriptions. *Proc. of the Int. Conference on Pattern Recognition and Artificial Intelligence, ICPRAI 2018*, pp. 707–711.

[2] Blei, D.M. (2014) Build, compute, critique, repeat: Data analysis with latent variable models. *Annual Review of Statistics and Its Application*, 1(1), 203–232.

[3] Chen, B. and Perona, P. (2016) Vision without the Image. *Journal of Sensors*, 16(4), 484.

[4] Desidera, G., Bertero, M., Boccacci, P., and Vicidomini, G. (2009) Image deblurring with Poisson data: From cells to galaxies. *Inverse Problems*, 25(12), 123006.

[5] Fossum, E.R., Teranishi, N., Theuwissen, *et al.*, (eds.) (2017) *Photon–Counting Image Sensors.* (MDPI, Basel).

[6] Goodman, J.W. (2015) *Statistical Optics, 2nd Ed.* (Wiley, New York).

[7] Gupta, M.R. and Chen, Y. (2011) Theory and use of the EM Algorithm. *Foundations and Trends in Signal Processing*, 4(3), 223–296.

[8] Hastie, T., Tibshirani, R., and Friedman, J. (2009) *The Elements of Statistical Learning, 2nd Ed.* (Springer, New York).

[9] Latecki, L.J., Lakmper, R., and Eckhardt, U. (2000) Shape descriptors for non-rigid shapes with a single closed contour. *Proc. IEEE*

Conf. Computer Vision and Pattern Recognition, CVPR, pp. 424–429.

[10] Mandel, L. and Wolf, E. (1995) *Optical Coherence and Quantum Optics*. (Cambridge University Press, UK).

[11] McLachlan, G.J. and Peel, D. (2000) *Finite Mixture Models*. (Wiley & Sons, Inc, New York, Chichester).

[12] Streit, R.L. (2010) Poisson Point Processes. Imaging, Tracking and Sensing. (Springer, New York).

© 2025 World Scientific Publishing Company
https://doi.org/10.1142/9789811267215_0009

Chapter 9

Nonlinear Mapping of Hyperspectral Data

Evgeny V. Myasnikov

Samara National Research University, Samara, the Russian Federation
mevg@geosamara.ru

The high spectral resolution of hyperspectral data allows us to solve important applied tasks with improved accuracy, but the costs of transmission, storage, processing, and recognition of such data are high. However, hyperspectral data contain substantial redundancy, so the important problem is to reduce the dimensionality of such data while preserving the improved quality of the solution of applied tasks.

In this chapter, to solve the above problem, we focus on a well-known nonlinear mapping technique, which is based on the principle of preserving pairwise Euclidean distances between vectors. We compare this technique to other dimensionality reduction techniques. To address the problem of the high computational complexity, we use a variety of methods, including stochastic gradient descent, interpolation, and space partitioning.

Next, we extend the considered technique in a natural way to use it with different spectral dissimilarity measures, such as spectral angle, spectral correlation, and spectral information divergence. In particular, several dimensionality reduction techniques based on the principle of preserving the selected pairwise dissimilarity measures are described in this chapter.

Finally, for hyperspectral images, we consider how the spatial information contained in images can be used in nonlinear mapping. In particular, we modify the dissimilarity measures to take into account the spatial context of image pixels using the order statistics.

All the experiments in this chapter are conducted using well-known open hyperspectral images.

Keywords: Hyperspectral image, nonlinear dimensionality reduction, nonlinear mapping, data mapping error, principal component analysis, Isomap, locally linear embedding, Laplacian eigenmaps, stochastic gradient descent, Space partitioning, lists of reference nodes, priority queues, interpolation, preserving mapping, spectral angle mapper, spectral correlation, spectral information divergence, spatial context, window functions, order statistics.

9.1 Introduction

A hyperspectral image (HS image) can be considered as a three-dimensional array having two spatial dimensions and one spectral dimension. HS images are captured by devices with a fine spectral resolution, so the spectral dimensionality of such images is usually high. Each pixel of an HS image is a vector containing up to several hundreds of spectral components.

This high spectral resolution of HS images provides a number of advantages over ordinary color and gray-scale images, as it allows us to detect materials (components) depicted in an image and solve important applied tasks (classification, target detection, etc.) with improved accuracy. But all these advantages come with the increased costs of transmission, storage, processing, and recognition of such images conditioned by the increased amount of data.

However, components (bands) in HS images are highly correlated, and such images contain substantial redundancy. So, the important problem is to reduce the redundancy of HS images while preserving the improved quality of the solution of applied tasks.

To reduce the redundancy of HS images, both feature selection and dimensionality reduction techniques were used and the latter group of techniques became more popular [16] generally as they do not require the creation of learning sets.

The dimensionality reduction techniques are usually divided into linear and nonlinear techniques. The linear techniques (principal

component analysis, independent component analysis, and projection pursuit) are widely used. But as HS images are subject to nonlinear effects [2], the nonlinear dimensionality reduction techniques have become more popular in the last years. The list of nonlinear techniques, which were used in the HS image analysis, includes nonlinear mapping (NLM) [35], Isomap [39], locally linear embedding (LLE) [34], Laplacian eigenmaps (LE) [3], and some other methods.

In this chapter, we focus on one of the oldest and well-known nonlinear dimensionality reduction techniques, the nonlinear mapping technique. In our preliminary experiments, this technique provided comparable or better results compared to other considered dimensionality reduction techniques. Besides, this technique can be extended in a natural way to be used with different spectral dissimilarity measures and to take into account the spatial context of an image.

The known disadvantage of the nonlinear mapping technique is its high computational complexity. In this chapter, we address this problem using a variety of methods, including stochastic gradient descent, interpolation, and space partitioning.

This chapter has the following structure. Section 9.2 gives a brief overview and describes the base nonlinear mapping technique. Section 9.3 is devoted to the methods, which allow improving the computational efficiency of the base method. Section 9.4 introduces the dissimilarity measures, which are used in HS image processing, and presents techniques based on the introduced measures. Section 9.5 describes approaches, which can be used to take into account the spatial information contained in HS images.

9.2 Nonlinear Dimensionality Reduction of HS Data

Nowadays, we can indicate a number of works in which nonlinear dimensionality reduction techniques have been successfully applied to the processing of multi- and HS images. For example, curvilinear component analysis (CCA) and curvilinear distance analysis (CDA) techniques have been applied to reduce the dimensionality of multispectral images in [10,15]. The Isomap technique was initially applied in [1, 2], and later in [6, 7]. The LLE technique was used in [11], and the same technique was used in [33] together with LE technique.

Later examples of the application of LLE, LE techniques, and their modifications can be found in papers [8,35,40–42]. The local tangent space alignment (LTSA) technique with modification was applied for HS image analysis in [34]. The NLM technique was applied, for example, in [18,20]. The citation list can be expanded.

In all these nonlinear techniques, it is required to measure the dissimilarity between pixels of an HS image. The most commonly used measure for this purpose is the Euclidean distance (ED). However, there are several other measures that have been used successfully in HS image analysis. The most widely used are the spectral angle (SAM) [13], spectral correlation [4], and spectral information divergence (SID).

Often, these measures have advantages over the ED in the field of HS image analysis; however, they have rarely been used in the field of nonlinear dimensionality reduction.

In particular, we can indicate some papers (for example, [1, 6, 7, 24, 40, 41]), in which the SAM measure was used with nonlinear dimensionality reduction techniques. In [1], the SAM measure was used together with the ED in the first stage of the ISOMAP method to determine neighbor pixels in the HS space. In [40], the authors study the effectiveness of the SAM measure and the ED to reduce dimensionality using Laplacian eigenmaps. In [41], the SAM measure was used to improve the Laplacian eigenmaps method to reduce the sensitivity to data skips in temporal sequences of multispectral satellite images. The paper [24] proposed several nonlinear mapping methods based on the principle of preservation of SAM.

We know only a few works [7,27], in which the spectral correlation measure or SID measure was used.

In spite of the fact that we can indicate a lot of papers in which spatial information is used in image analysis, in particular, in the analysis of HS images (a recent review on this topic can be found in [39]), the number of papers devoted to the problem of using spatial information of HS images in nonlinear dimensionality reduction techniques is rather small (see, for example, [5, 23, 26, 34]).

9.2.1 *Nonlinear mapping algorithm*

Let us consider the HS image X having the width W and height H. This image contains $N = WH$ image pixels. In this chapter, the spatial coordinates of the pixel often are not essential to us, and we

denote image pixels as x_i, $i = 1 \ldots N$ assuming that some scanning sequence exists (for example, progressive scan), which gives a meaningful/reasonable value to the index i. Each pixel of the image X can be considered as a vector in the M-dimensional HS space R^M.

The purpose of a dimensionality reduction procedure consists of calculating the coordinates y_i, $i = N$ of all data points of a dataset in some lower dimensional space R^m, where $m < M$. In our case, the data points are pixels x_i of the HS image X.

The idea of the nonlinear mapping algorithm described in the following consists in the preserving of the structure of data. Here we assume that the structure is defined by pairwise distances between all the points of the dataset, and usually, the ED $d()$ is used.

In degenerate cases, the points x_i reside in the lower dimensional subspace R^m of the multidimensional space R^M, and all the pairwise distances $d(x_i, x_j)$ can be preserved accurately so that $d(y_i, y_j) = d(x_i, x_j)$, $i, j = 1 \ldots N$. But in practice, the distances $d(x_i, x_j)$ can be preserved only approximately, and we can measure the total error induced by a dimensionality reduction procedure as follows:

$$\varepsilon = \sum_{i,j=1(i<j)}^{N} (d(x_i, x_j) - d(y_i, y_j))^2.$$

To give more weights to desirable distances, the weight constants $\rho_{i,j}$ are introduced, as long as a normalizing factor μ. So, the data mapping error can be estimated by the equation

$$\varepsilon = \mu \cdot \sum_{i,j=1(i<j)}^{N} (\rho_{i,j}(d(x_i, x_j) - d(y_i, y_j))^2). \tag{9.1}$$

For example, for the following normalizing factor and unary weights

$$\mu = 1 \Big/ \sum_{i<j} d^2(x_i, x_j), \quad \rho_{i,j} = 1,$$

equation (9.1) leads to Kruskal stress [14].

If we would like to give more attention to small distances

$$\mu = 1 \Big/ \sum_{i<j} d(x_i, x_j), \quad \rho_{i,j} = 1/d(x_i, x_j),$$

the same equation (9.1) leads to the Sammon data mapping error [32].

After the definition of the error function (9.1), the dimensionality reduction problem can be stated as a minimization problem

$$\varepsilon \underset{Y}{\longrightarrow} \min, \quad Y = (y_1, \ldots, y_N). \tag{9.2}$$

To solve problem (9.2), a number of optimization techniques can be used, but the gradient descent approach is used most often.

The base version of the gradient descent algorithm assumes iterative changing the parameters Y using the following equation:

$$Y(t+1) = Y(t) - \alpha \nabla \varepsilon, \tag{9.3}$$

where t is the number of the iteration, $\nabla\varepsilon$ is the gradient of the objective function, and α is the coefficient of the gradient descent (step size).

After finding all the partial derivatives of the objective function (9.1), the iterative optimization algorithm takes the following form:

$$y_i(t+1) = y_i(t) + 2\alpha\mu \sum_{j=1(j\neq i)}^{N} \rho_{i,j} \frac{d(x_i, x_j) - d(y_i, y_j)}{d(y_i, y_j)} \cdot (y_i(t) - y_j(t)). \tag{9.4}$$

So to obtain the final solution to problem (9.2), one needs to initialize the coordinates $y_i(0)$, $i = 1 \ldots N$ of data points in the output low-dimensional space and iteratively change the coordinates $y_i(t)$ according to the above expression until the coordinates become stable.

9.2.2 *Numerical experiments*

In this section, we provide a numerical comparison of the described above nonlinear mapping algorithm and some alternative techniques. For the reported study, we used two well-known HS image scenes [9], which were supplied with groundtruth segmentation: Indian pines and Kennedy Space Center. Both HS image scenes were acquired using the AVIRIS sensor.

The first scene contains 145×145 pixels and 224 spectral bands. In our experiments, we used the image containing 204 bands, in which some bands were discarded due to high noise and water absorption. The classified pixels of the groundtruth image are divided into 16 classes.

Kennedy Space Center scene contains 512×614 pixels. The version containing 176 spectral bands was used in the experiments. The groundtruth image contains information only on a small number of pixels divided into 13 classes.

The list of studied techniques includes the linear principal component analysis (PCA) technique, the described nonlinear mapping algorithm (NLM), and the following nonlinear dimensionality reduction techniques: Isomap (LmIsomap) [36, 37], locally linear embedding (LLE) [31], and Laplacian eigenmaps (LE) [3].

Considering that the most frequently solved problem is the per-pixel classification task, and the listed techniques are based on different principles, to compare these techniques, we measured the quality of per-pixel classification using the 1-NN classifier. To measure the quality of classification, we used the overall classification accuracy, defined as the proportion of the correctly classified pixels of the test set.

So the whole set of ground truth samples was divided into a training (60%) and a test (40%) subset in our experiments. The dimensionality of the reduced space ranged from 3 to 30.

To perform the experiments, we used PCA implementation provided with Matlab and C^{++} implementation of nonlinear mapping algorithm. For LLE, LE, and Isomap, we used Matlab Toolbox for Dimensionality Reduction [38]. The results of the experiments are shown in Figure 9.1.

As can be seen from the figure, in all the considered cases, the best results were obtained using the nonlinear mapping technique.

It is worth noting that three graph-based nonlinear dimensionality reduction techniques (Isomap, LLE, and Laplacian eigenmaps) have the mataparameter, which defines the number of neighbors of a graph vertex. Nevertheless, we were not able to outperform the nonlinear mapping by varying this parameter in the reasonable range from 10 to 100 with the step equal to 10. The results depicted in the above figures are shown for the default value equal to 12.

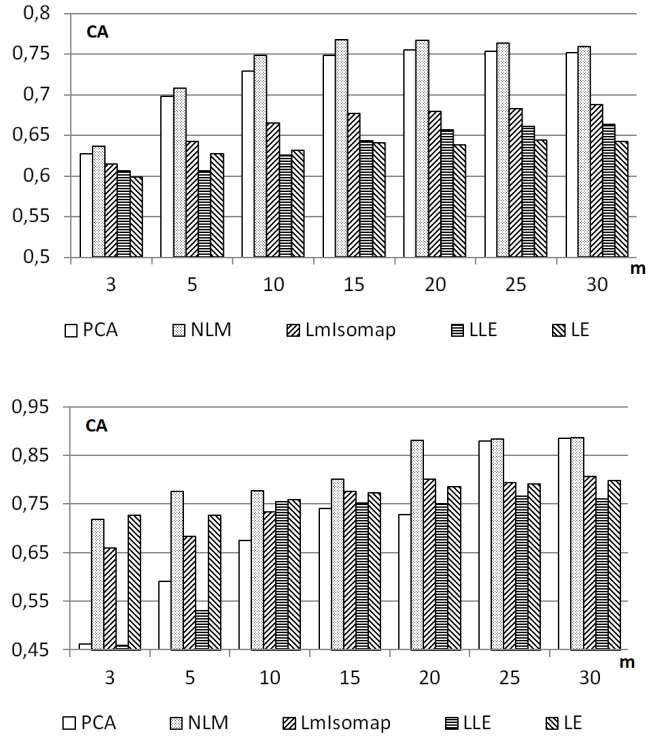


Figure 9.1. The dependence of the classification accuracy (CA) on the dimensionality of the reduced space (m) for PCA and nonlinear dimensionality reduction techniques: Indian Pines (top) and Kennedy Space Center (bottom) hyperspectral images.

Despite the difference in the implementation of these methods, it was obvious that nonlinear methods are incomparably slower than the PCA. For that reason, in fact, we used the Landmark Isomap method [37], which is a faster version of Isomap, and SGD version of the nonlinear mapping algorithm, which is described in the following section.

9.3 Fast Nonlinear Mapping Methods

Unfortunately, the basic approach described in the previous section suffers from drawbacks associated with the high computational complexity of this algorithm. As the optimization procedure is iterative and the total number of iterations depends on many factors including

the step size α (or the adaptive scheme of its change), stopping criterion, etc., it makes sense to estimate the computational complexity (or run time) for one iteration of the optimization procedure.

According to (9.4) for each data point x_i, $i = 1 \ldots N$, we calculate the impact from all other points x_j, $j = 1 \ldots N$ $(i \neq j)$. So the calculation time grows as the square of the number of data points (or pixels, in the case of HS image processing). Then the complexity is linear on the dimensionality of input and output spaces as EDs in both spaces take part in the calculations.

As the distances $d(x_i, x_j)$ in the multidimensional space remain constant in the optimization process, it may seem reasonable to calculate them before iterative optimization. But storing precalculated distances requires $O(N^2)$ memory, and it takes $O(N^2)$ operations to calculate such a distance matrix. Unfortunately, for large datasets, it is practically impossible. For example, it requires approximately 16 GB of operational memory to store half of the distance matrix (symmetric) for an image containing 256×256 pixels.

Due to the above reasons, the nonlinear mapping algorithm in the described above simple version cannot be used to process HS images. To overcome these drawbacks, in this section, we describe some approaches to speed up the calculations.

9.3.1 *Stochastic gradient descent*

The stochastic gradient descent algorithm is widely used in the case of large-scale problems. A simple version of this algorithm uses a single random example to approximate the gradient at each iteration. A widely used modification of this algorithm uses small subsets of randomly selected points (mini-batches) that get smoother convergence.

In our case, we can approximate gradient in equation (3) using the set r containing indexes r_j, $0 < r_j < N$, $j = 1 \ldots R$ of randomly selected pixels at each iteration. Thus, for the stochastic gradient descent based on mini-batches, equation (9.4) takes the form [20]

$$y_i(t+1) = y_i(t) + 2\alpha\mu \sum_{j=1(r_j \neq i)}^{R} \rho_{i,r_j} \frac{d(x_i, x_{r_j}) - d(y_i, y_{r_j})}{d(y_i, y_{r_j})} \cdot (y_i(t) - y_{r_j}(t)). \tag{9.5}$$

Here the computational complexity is reduced to $O(RN)$, and usually $R \ll N$ in this algorithm.

The described algorithm gives the opportunity to process small fragments of HS images and provides quite good results in our experiments. Several further improvements of the stochastic gradient descent algorithm, namely, the momentum method and adaptive gradient algorithm, are studied in [20] for the nonlinear mapping of HS data.

9.3.2 *Space partitioning*

Another idea to speed up the calculations in the context of the considered task has been originated from physics. According to this approach, we perform the partitioning of space into cells and consider "point to cell" interactions instead of "point to point" interactions when possible. This idea of acceleration is widely used in simulations to solve the n-body problem. In our case, the idea of approximating the "point to point" interactions by the "point to cell" interactions allows us to accelerate the iterative optimization process significantly.

Let us assume that all the data points x_i, $i = 1 \dots N$, are distributed across the subsets s_j, $j = 1 \dots S$, and s_j^k is the index of the kth point from the jth subset in the whole dataset: $0 < s_j^k < N$. Then (9.4) can be rewritten as [17]

$$y_i(t+1) = y_i(t) + 2\alpha\mu \sum_{j=1}^{S} \sum_{k=1\left(s_j^k \neq i\right)}^{|s_j|} \times \rho_{i,s_j^k} \frac{d\left(x_i, x_{s_j^k}\right) - d\left(y_i, y_{s_j^k}\right)}{d\left(y_i, y_{s_j^k}\right)} \cdot \left(y_i(t) - y_{s_j^k}(t)\right). \tag{9.6}$$

If the subsets s_j, $j = 1 \dots S$, are compact enough (well separated) with relation to the ith data point, then the centroids of the subsets are defined as average values of coordinates of data points contained in these subsets

$$x_{s_j} = \frac{1}{|s_j|} \sum_{k=1}^{|s_j|} x_{s_j}, \quad y_{s_j} = \frac{1}{|s_j|} \sum_{k=1}^{|s_j|} y_{s_j}$$

can be used to approximate the above equation (9.6) and, therefore, equation (9.4):

$$y_i(t+1) = y_i(t) + 2\alpha\mu \sum_{j=1}^{S} \sigma_{i,j}(t),$$

$$\sigma_{i,j}(t) = \begin{cases} |s_j| \rho_{i,s_j} \dfrac{d(x_i, x_{s_j}) - d(y_i, y_{s_j})}{d(y_i, y_{s_j})} \cdot \left(y_i(t) - y_{s_j}(t)\right), \\ \qquad\qquad \text{if } w(i, s_j) = 1 \\ \displaystyle\sum_{k=1\left(s_j^k \neq i\right)}^{|s_j|} \rho_{i,s_j^k} \dfrac{d(x_i, x_{s_j^k}) - d(y_i, y_{s_j^k})}{d\left(y_i, y_{s_j^k}\right)} \cdot \left(y_i(t) - y_{s_j^k}(t)\right), \\ \qquad\qquad \text{otherwise} \end{cases}.$$

Here, $w(i, s_j)$ is the function (decomposition criterion), which takes a value equal to one, if the subset s_j is compact enough (well separated from the ith data point), and zero value otherwise.

This approximation induces some error in the optimization, which depends on the distribution of points across the subsets.

Assuming that the average number of summands is constant (subsets s_j do not change in the optimization process) and is equal to Q:

$$Q = \frac{1}{N} \sum_{i=1}^{N} \sum_{j=1}^{S} (w(i, s_j) + |s_j|(1 - w(i, s_j))), \tag{9.7}$$

we can estimate the computational complexity as $O(QN)$.

9.3.2.1 *Input and output space partitioning using space partitioning data structures*

The data dimensionality reduction problem is in contrast to the n-body problem as the above-mentioned subsets can be formed based on the characteristics of data points in two spaces: in the input multidimensional space and in the output low-dimensional space. On the one hand, one should take into account the configuration of data points in both input and output spaces to control the approximation error, as distances in both spaces are approximated. On the other hand, we deal with the nonlinear mapping technique, which is based on the principle of preserving the pairwise distances between points.

Therefore, if some subset is compact enough in the multidimensional input space, then we can hope that in the output low-dimensional space, data points will also be located quite compactly, and the subset center in the output space will also be a good representative of data points contained in it.

Some research on the considered issue carried out in [22] shows that the partitioning of the input space is preferable to the partitioning of the output space. The method that takes into account both configurations in input and output spaces allows us to achieve more accurate approximations with higher computational costs, and it requires selecting parameters carefully. The effect on the final data representation error for this technique is not significant.

After determining the space in which the partitioning will be performed, it is necessary to determine how the partitioning will be performed. It is known that the use of uniform grids for partitioning the multidimensional space is inefficient. So we would like to have some adaptive procedure, which could carry out a hierarchical partitioning for a large dataset.

On the one hand, the most natural way to group points in a multidimensional space according to their characteristics is clustering. On the other hand, the computational complexity even in the case of effective agglomerative clustering is $O(N^2)$, and it is more reasonable to use space partitioning trees that can be built in less time. An evaluation of three different space partitioning trees, namely, kd-tree, vp-tree, and cluster tree, is described in [20]. The study shows that space partitioning data structures like vp-trees and cluster trees can be successfully applied in multidimensional spaces for the considered problem.

9.3.2.2 *Lists of reference nodes*

In the case of partitioning the input space, we can form the above-mentioned subsets s_j, $j = 1 \dots S$, once before the optimization process. This is possible as the space partitioning tree constructed in the input space is constant in the optimization process, and the multidimensional input coordinates of data points x_i, $i = 1 \dots N$, are also constant.

To construct the above-mentioned subsets, for each data point x_i, the algorithm traverses the constructed space partitioning tree

starting from the top-level node. In each considered node, the algorithm makes a decision whether the associated set of points s is well separated from the point x_i (the decomposition criterion $w(i, s_j) = 1$) or not. If so, we add the set s as a whole to the interaction list of the point x_i. Otherwise ($w(i, s_j) = 0$), we traverse the child nodes of the node s of the constructed space partitioning structure. As a result, for each point x_i, we form an interaction list (so-called list of reference nodes) containing well-separated subsets of points or data points. A detailed description of the above algorithm can be found in [17].

It should be noted that the length of formed lists defines the computational complexity of the optimization process. The length itself depends on several factors: point distribution, space partitioning tree used, decomposition criterion, etc. As a result, it is difficult to predict the length and, therefore, to adjust the computational complexity of the optimization process. So the experiments needed to choose the parameters of the decomposition criterion and to obtain the desirable average length of lists.

9.3.2.3 *How to adjust computational complexity*

The stochastic gradient descent algorithm described above allows adjusting the computational complexity directly by varying the number of elements in mini-batches. To obtain this useful property for the methods based on space partitioning techniques, an approach based on priority queues was suggested in [18]. The main idea of this approach consists of using the priority queues of a length constrained by the predefined value.

In this case, we define some separability measure, which shows how well the considered node of a partitioning tree is separated from the point. This measure is used as a priority in the queue so that the most badly separated nodes are first, followed by better-separated subsets.

The algorithm for the queue construction starts with the only element representing the root of the space partitioning tree and is followed by the consequent decomposition of the first element (most badly separated) until the length of the queue is less than the predefined value Q.

It is worth noting that the described scheme allows not only to reduce the number of summands by approximating subsets by their centers but also to make the optimization process independent of the dimensionality of the input space. The latter requires precalculating and storing the input distances from data points to centers of the subsets used in approximation (reference nodes). This requires $O(QN)$ memory.

9.3.3 *Interpolation*

Another way to address the problem of high computational complexity for the considered task is the use of interpolation methods. In this case, we map K data points from multidimensional to low-dimensional space using one of the considered above methods and then sequentially map the remaining $(N - K)$ data points using one of the multivariate interpolation methods. Several multivariate interpolation methods, including nearest neighbor interpolation, inverse distance weighting, radial basis functions, and interpolation by data mapping error minimization, were evaluated in [21] for the considered task. It was shown that the interpolation by radial basis functions and the interpolation by the data mapping error minimization provide better results. The latter algorithm inherits the iterative nature of the gradient descent algorithm and requires $O(IK(N - K))$ operations to map the remaining $(N - K)$ data points that are compatible with stochastic gradient descent (I is the number of iterations). Other considered methods require $O(K(N - K))$ operations to map remaining points, and besides that, the interpolation by radial basis functions requires solving a system of linear equations that can be done in $O(K^3)$ operations.

9.3.4 *Numerical experiments*

To illustrate the described acceleration techniques, we conducted experiments with Indian Pines HS image. In particular, we performed nonlinear mapping into three-dimensional space (3D mapping) using the following techniques: the stochastic gradient descent (SGD), the technique based on the space partitioning and priority queues (PQ), and the interpolation technique based on the radial basis functions (SGD+RBF).

For the SGD, we used mini-batches containing $R = 25$ and $R = 50$ pixels. For the PQ, we used vantage point trees (vp-trees) as a binary space partitioning data structures. The formed priority queues contained up to $Q = 100$ and $Q = 200$ elements (reference nodes). For the SGD+RBF, we used base sets containing $K = 500$ and $K = 1000$ randomly chosen points. The solution for the base sets was obtained using SGD with $R = 50$. The rest $(N - K)$ points were mapped using the RBF functions.

The results of the experiments are shown in Figure 9.2.

As can be seen, the best values of the nonlinear mapping quality were provided by SGD and PQ techniques. For this reason, we avoid interpolation when working with small fragments or images.

The PQ provided low nonlinear mapping error in minimal time as the algorithm allows precalculating distance in hyperspectral space. Nevertheless, in our further experiments, we will use the SGD approach, as the PQ technique relies heavily on the ED as

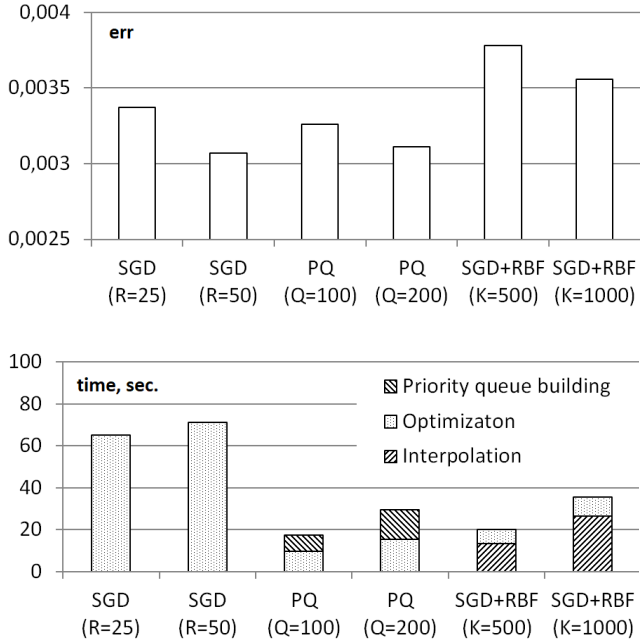


Figure 9.2. Experimental results for different acceleration techniques: the data mapping error (top) and the processing time (bottom) for the Indian Pines hyperspectral image.

a metric, and the following section is devoted to alternative dissimilarity measures.

9.4 Dissimilarity Measures in Nonlinear Mapping

Up to the moment, we assumed that the ED is used as a dissimilarity measure. As it was outlined earlier, there are several other popular dissimilarity measures widely used in the field of HS image analysis. These measures are SAM, spectral correlation, SID, and some other measures. In this section, we show how the nonlinear mapping algorithm described above can be applied with non-Euclidean measures.

9.4.1 *Preserving non-Euclidean dissimilarities*

The most well-known non-Euclidean measure often used with HS images is the spectral angle mapper (SAM) measure [13]:

$$\theta(x_i, x_j) = \arccos\left(\frac{x_i^T x_j}{\|x_i\| \, \|x_j\|}\right). \tag{9.8}$$

Here, x_i and x_j are two vectors corresponding to ith and jth pixels in an HS image. Thus, the SAM measures the spectral dissimilarity as the angle between two spectral signatures x_i and x_j in amplitude independent manner that gives some independence to the variations in illumination.

In the considered approach, we introduce the SAM preserving error in the following form:

$$\varepsilon = \mu \sum_{i,j=1(i<j)}^{N} (\theta(x_i, x_j) - \theta(y_i, y_j))^2. \tag{9.9}$$

As this function is differentiable (except points $\|x_i\| = 0, \|y_i\| = 0$), we can apply the gradient descent algorithm to obtain a suboptimal solution.

The latter requires calculating the gradient of the SAM preserving error. Finally, the iterative equation for the coordinates of data points

in the output space takes the following form [24]:

$$y_{ik}(t+1) = y_{ik}(t) + 2\alpha\mu \sum_{j=1(j\neq i)}^{N} (\theta(x_i, x_j) - \theta(y_i, y_j)) \times \frac{y_{jk} - y_{ik}(y_i^T y_j)/\|y_i\|^2}{\sqrt{\|y_i\|^2\|y_j\|^2 - (y_i^T y_j)^2}}. \quad (9.10)$$

As this procedure performs spectral angle preserving mapping, and the output vector lengths do not affect the mapping quality (9.9), we can consider the special case $\|y_i\| = 1$. In this case, expression (9.10) takes the following form:

$$y_{ik}(t+1) = y_{ik}(t) + 2\alpha\mu \sum_{j=1(j\neq i)}^{N} (\theta(x_i, x_j) - \theta(y_i, y_j)) \times \frac{y_{jk} - y_{ik}(y_i^T y_j)}{\sqrt{1 - (y_i^T y_j)^2}}. \quad (9.11)$$

As all the vectors' lengths are equal to one in the considered case, then it is possible to convert features from Cartesian coordinates to hyperspherical coordinates and to truncate the radial component. Thus, the considered method performs nonlinear mapping onto the unit hypersphere using the SAM preserving principle.

It was mentioned above that other techniques can be used to solve the considered problem. An alternative way considered in [24] is based on similar principles, but the solution is derived in a hyperspherical coordinate system (in contrast to Cartesian coordinates). Since the optimization procedures in both methods are more difficult compared to the base version, there were also suggested simpler alternatives in [24].

According to the considered approach, spectral information divergence preserving mapping [28] can be introduced by the formulae:

$$y_{ik}(t+1) = y_{ik}(t) + 2\alpha\mu \cdot \sum_{j=1(j\neq i)}^{N} \frac{\mathrm{SID}(x_i, x_j) - \mathrm{SID}(y_i, y_j)}{\sum_{l=1}^{m} y_{il}} \times \left(1 - s_{ij} - \frac{p_n(y_j)}{p_n(y_i)} + \log\frac{p_n(y_i)}{p_n(y_j)}\right),$$

where SID() is the spectral information divergence measure,

$$p_k(x_i) = \frac{x_{ik}}{\sum_{l=1}^{M} x_{il}}, \quad s_{ij} = \sum_{k=1}^{m} \left(p_k(y_i) + p_k(y_i) \log \frac{p_k(y_i)}{p_k(y_j)} - p_k(y_j) \right).$$

Several other methods developed by the author, include spectral correlation preserving mapping [27] and Hellinger divergence preserving mapping [29]. The latter constructs output space based on the Hellinger divergence, which exhibits the true metric properties. Hence, binary space partitioning trees can be used to accelerate classification queries [29].

9.4.2 *Approximation of dissimilarities by Euclidean distances*

Another approach to embedding non-Euclidean dissimilarity measures into nonlinear mapping algorithm consists of the approximation of given dissimilarities by EDs in a lower-dimensional space. This approach is in good agreement with the idea of multidimensional scaling.

The implementation of this approach comes down to replacing the source EDs $d(x_i, x_j)$ in the error function (9.1) and optimization algorithm (9.4) with the required dissimilarity measure.

The efficiency of such a naïve approach depends on how well the dissimilarity measures can be approximated by EDs in lower dimensions. For the SAM, it is a known fact that the distance between two points on the unit hypersphere is equal to the angle between corresponding vectors. This fact allows us to hope that EDs can be good approximations of SAMs. This assumption was confirmed in [24] (SAED method) as it was shown that this simple approach is almost equal in accuracy to the considered above (9.11).

But for the spectral correlation measure, the same approach gave noticeably worse results. To convert spectral correlation $r(x_i, x_j)$ to dissimilarity measure, we used simple transformation: $\delta(x_i, x_j) = (1 - r(x_i, x_j))$. In our experiments [27], the approximation of spectral correlation measures by EDs provided significantly greater errors compared to the preserving of spectral correlation measures. Due to the lack of theoretical results, experiments are needed to estimate the applicability of the simple approach to other measures.

9.4.3 *Numerical experiments*

We conducted experimental studies of the described nonlinear mapping techniques based on different dissimilarity measures. In particular, in the dimensionality reduction stage, we used the nonlinear mapping technique based on Euclidean distances (NLM), the technique based on the approximation of spectral angles by Euclidean distances (SAED), the spectral correlation preserving mapping (SCPM), and the spectral information divergence preserving mapping (SDPM). All listed methods were implemented using the stochastic gradient-based approach.

In the classification stage, we used 1-NN classifiers based on measures, which correspond to the used dimensionality reduction techniques. For the NLM and SAED methods, we used the classifier based on ED. For the SCPM method, we used the measure calculated as one minus the correlation. For the SDPM, we used SID.

In our experiments, we used previously described Indian Pines and Kennedy Space Center HS images. For both images, we applied the groundtruth masks and selected classified pixels for the dimensionality reduction stage. We varied output dimensionality from $m = 3$ to $m = 30$ and measured the classification accuracy. The results are shown in Figure 9.3.

As you can see, in most cases, the considered methods showed acceptable values of classification accuracy. The best quality was provided by SAM-based and SID-based approaches. The worst results were provided by SCPM for low output dimensionalities.

9.5 Using the Spatial Context

Up to the moment, all the considered dimensionality reduction techniques operated only in the spectral space of HS images. The spatial information contained in HS images was not taken into account.

Nevertheless, today, it is common to use spectral-spatial techniques in HS image analysis, especially in classification. We can indicate two main techniques, which are used to take into account the spatial information of HS images. The first technique consists of using the kernel function, which operates both in spectral and spatial spaces. The second technique consists of constructing

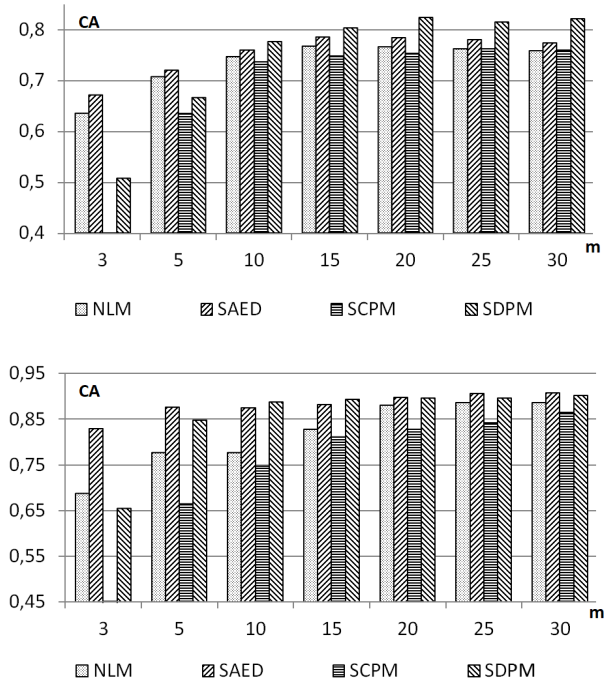


Figure 9.3. The dependence of the classification accuracy (CA) on the dimensionality of the reduced space (m) for nonlinear mapping techniques based on different dissimilarity measures: Indian Pines (top) and Kennedy Space Center (bottom) hyperspectral images.

similarity/dissimilarity measures, which take into account both the spatial context and spectral features of image pixels. The latter technique can be used with nonlinear mapping algorithms in a quite natural way.

New dissimilarity measures can exploit different schemes of embedding spatial context. These measures also can be based on different underlying dissimilarity measures, such as ED or SAM. In this section, we consider two schemes of embedding spatial context, described in detail in the following.

Recent advances in the field include a feature fusion framework for dimensionality reduction [30], taking into account, for example, spectral and local texture features, which is however beyond the scope of this chapter.

9.5.1 *Window functions*

The first scheme of embedding spatial context consists in extending of the feature space by values of neighbor pixels. The well-known implementation of this approach in per-pixel image classification and segmentation uses 4- or 8-pixel neighborhood. In a more general case, we consider spatial pixel neighborhood with some radius R.

On the one hand, the dimensionality reduction procedure does not use any groundtruth information, and we cannot build any procedure, which assigns greater weights to more informative pixels. On the other hand, it is intuitively clear that closer pixels should have a greater impact on the dissimilarity than the further ones. It is also clear that in the absence of prior information about the depicted scene, these weights should not depend on the spatial direction.

These considerations lead us to the use of two-dimensional radial-symmetric window functions as weighting functions. In this case, the dissimilarity measure ρ between two image pixels x_i and x_j with neighbor pixels

$$x_{i_1}, x_{i_2}, \dots x_{i_K}, \quad \text{and} \quad x_{j_1}, x_{j_2}, \dots x_{j_K}$$

can be expressed as follows:

$$\rho(x_i, x_j) = \left(\sum_{k=1}^{K} w_k \sum_{l=1}^{m} (x_{i_k l} - x_{j_k l})^2 \right)^{1/2},$$

where each weighting coefficient w_k is defined by the value of the window function $w(r)$, and its argument r is the ED in the spatial domain between the pixel x_i and kth pixel from its neighborhood:

$$w_k = w(r(x_i, x_{i_k})).$$

9.5.2 *Order statistics*

While the scheme described above operates in a non-adaptive manner, assigning equal weights to equally distant pixels in a spatial neighborhood, it makes sense to use adaptive weighting, based on the similarity of the pixels in a neighborhood. This idea was implemented in [23, 26] in the scheme using the order statistics of a pixel neighborhood.

This scheme consists of the sorting of all the pixels

$$x_{i_1}, x_{i_2}, \dots x_{i_K}$$

in a spatial neighborhood by proximity to the central pixel x_{i1}:

$$\delta(x_{i_1}, x_{(i_1)}) \le \delta(x_{i_1}, x_{(i_2)}) \le \dots \le \delta(x_{i_1}, x_{(i_K)}).$$

Here, the reordered set of pixels

$$x_{(i_1)}, x_{(i_2)}, \dots x_{(i_K)}$$

consists of order statistics and δ is a measure of dissimilarity between image pixels in the hyperspectral space.

Now we can truncate this set and use some number $S \leq= K$ of order statistics in the dissimilarity measure:

$$\rho(x_i, x_j) = \eta \sum_{s=1}^{S} w_s \delta\left(x_{(i_s)}, x_{(j_s)}\right).$$

Here, we use inverse weighting $w_s = 1/s$, and S is the number of the order statistics used as a spatial context.

The above equation is ready to use with non-Euclidean dissimilarity measures, as any non-negative measure can be used as a dissimilarity δ.

9.5.3 *Numerical experiments*

In this section, we show some experimental results for the scheme described above based on order statistics. In particular, we consider two values of the neighborhood radius: $R = 1$, and $R = 2$, and minimum ($S = 2$) and maximum ($S = 5$ for $R = 1$, and $S = 13$ for $R = 2$) number of order statistics. We provide results both for ED and SAM dissimilarity measures. For the ED-based version, we also show results of the base NLM technique marked as "No context" (equivalent for $S = 1$). For the SAM-based version, we show results for the SAED method described earlier. The latter is also marked as "No context" (equivalent for $S = 1$). For the comparison purposes, we show results for the linear PCA technique. The experiments were conducted for the Indian Pines HS image. In the

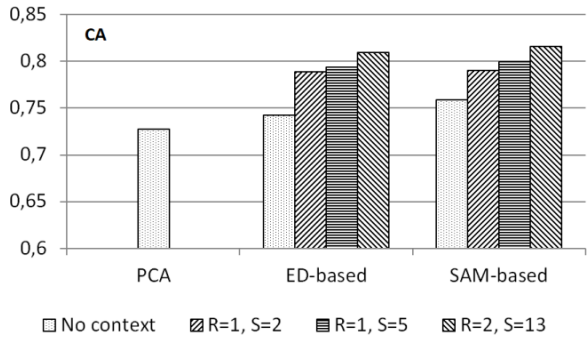


Figure 9.4. The dependence of the classification accuracy (CA) on the radius of the neighborhood (R) and the number of order statistics (S) used as a spatial context for Euclidean and spectral angle-based nonlinear mapping.

dimensionality reduction stage, we used all of the image pixels. The output dimensionality was set to $m = 5$.

As you can see in Figure 9.4, the use of the first-order statistic ($S = 2$) allowed improving the classification quality by several percent compared to base versions (NLM or SAED). The use of all the order statistics allowed getting even better results that were unattainable without spatial context even with higher output dimensionalities.

The results for the scheme based on window functions, as well as the study for other parameter values and HS image, can be found in [23, 26].

9.6 Conclusion

The HS image analysis is a relatively new research area. We believe that interest in this area of research will grow with the emergence of affordable HS imagers and wider distribution of hyperspectral imaging.

While HS images provide many advantages over grayscale and color images, dimensionality reduction techniques should be used to reduce the redundancy of HS images and, therefore, to reduce processing costs. The nature of HS images assumes that nonlinear techniques should be used, as different nonlinear effects take place in the process of their formation.

In this chapter, we described some nonlinear mapping techniques, united by the idea of preserving pairwise dissimilarities between image pixels. In the general nonlinear mapping framework, we showed how new dissimilarity measures can be used and how the spatial context of HS images can be embedded. Considerable attention has been paid to increasing the speed of the nonlinear mapping algorithm. In general, we position the considered framework as an expandable toolkit for HS image analysis.

References

[1] Bachmann, C.M., Ainsworth, T.L., and Fusina, R.A. (2005) Exploiting manifold geometry in hyperspectral imagery. *IEEE Transactions on Geoscience and Remote Sensing*, 43(3), 441–454.

[2] Bachmann, C.M., Ainsworth, T.L., and Fusina, R.A. (2006) Improved manifold coordinate representations of large-scale hyperspectral scenes. *IEEE Transactions on Geoscience and Remote Sensing*, 44(10), 2786–2803.

[3] Belkin, M. and Niyogi, P. (2001) Laplacian eigenmaps and spectral techniques for embedding and clustering. *Advances in Neural Information Processing Systems*, 14, 586–691.

[4] Blumenson, L.E. and Bakker, W. (1997) CCSM: Cross correlogram spectral matching. *International Journal of Remote Sensing*, 18(5), 1197–1201.

[5] Borhani, M. and Ghassemian, H. (2015) Kernel multivariate spectral-spatial analysis of hyperspectral data. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 8(6), 2418–2426.

[6] Ding, L., Tang, P., and Li, H. (2013) Dimensionality reduction and classification for hyperspectral remote sensing data using ISOMAP. *Hongwai yu Jiguang Gongcheng/Infrared and Laser Engineering,* 6(10), 2707–2711.

[7] Du, P., Wang, X., Tan, K., and Xia, J. (2011) Dimensionality reduction and feature extraction from hyperspectral remote sensing imagery based on manifold learning. *Wuhan Daxue Xuebao (Xinxi Kexue Ban)/Geomatics and Information Science of Wuhan University*, 36(2), 148–152.

[8] Hong, D.F., Yokoya, N., and Zhu, X.X. (2016) Local manifold learning with robust neighbors selection for hyperspectral dimensionality reduction. *Proceedings of International Geoscience and Remote Sensing Symposium (IGARSS)*, pp. 40–43.

[9] Hyperspectral Remote Sensing Scenes. http://www.ehu.eus/ccwintco/index.php?title=HyperspectralRemoteSensingScenes.

[10] Journaux, L., Foucherot, I., and Gouton, P. (2004) Nonlinear reduction of multispectral images by curvilinear component analysis: Application and optimization. *Proceedings of International Conference on CSIMTA 2004*.

[11] Kim, D.H. and Finkel, L.H. (2003) Hyperspectral image processing using locally linear embedding. *Proceedings of First International IEEE EMBS Conference on Neural Engineering*, pp. 316–319.

[12] Kong, X., Shu, N., Huang, W., and Fu, J. (2010) The research on effectiveness of spectral similarity measures for hyperspectral image. *Proceedings of 3rd International IEEE Congress on Image and Signal Processing, CISP2010,* pp. 2269–2273.

[13] Kruse, F.A., *et al.* (1993) The spectral image processing system (SIPS) interactive visualization and analysis of imaging spectrometer data. *Remote Sensing of Environment*, 44, 145–163.

[14] Kruskal, J.B. (1964) Multidimensional scaling by optimizing goodness of fit to a non-metric hypothesis. *Psychometrika*, 29, 1–27.

[15] Lennon, M., Mercier, G., Mouchot, M., and Hubert-Moy, L. (2002) Curvilinear component analysis for nonlinear dimensionality reduction of hyperspectral images. *Proceedings of Image and Signal Processing for Remote Sensing VII*, Vol. 4541 (SPIE), pp. 157–168.

[16] Lunga, D., Prasad, S., Crawford, M., and Ersoy, O. (2014) Manifold-learning-based feature extraction for classification of hyperspectral data. *IEEE Signal Processing Magazine*, 31(1), 55–66.

[17] Myasnikov, E.V. (2012) A nonlinear method for dimensionality reduction of data using reference nodes. *Pattern Recognition and Image Analysis*, 22(2), 337–345.

[18] Myasnikov, E.V. (2015) Nonlinear mapping methods with adjustable computational complexity for hyperspectral image analysis. *Proceedings of 8th International Conference on Machine Vision*, Vol. 9875 (SPIE), p. 987508.

[19] Myasnikov, E.V. (2015) Evaluation of space partitioning data structures for nonlinear mapping. *Proceedings of 23rd International Conference in Central Europe on Computer Graphics, Visualization and Computer Vision — Full Papers Proceedings*, Vol. 2015 (WSCG), pp. 109–118.

[20] Myasnikov, E.V. (2016) Evaluation of stochastic gradient descent methods for nonlinear mapping of hyperspectral data. *Proceedings of Image Analysis and Recognition*. LNCS, Vol. 9730, pp. 276–283.

[21] Myasnikov, E.V. (2016) The use of interpolation methods for nonlinear mapping. *Proceedings of Computer Vision and Graphics*. LNCS, Vol. 9972, pp. 649–655.

[22] Myasnikov, E.V. (2016) Analysis of approaches to feature space partitioning for nonlinear dimensionality reduction. *Pattern Recognition and Image Analysis*, 26(3), 474–482.
[23] Myasnikov, E.V. (2017) Exploiting spatial context in nonlinear mapping of hyperspectral image data. *Proceedings of Image Analysis and Processing.* LNCS, Vol. 10485, pp. 180–190.
[24] Myasnikov, E. (2017) Nonlinear mapping based on spectral angle preserving principle for hyperspectral image analysis. *Proceedings of Computer Analysis of Images and Patterns*. LNCS, Vol. 10425, pp. 416–427.
[25] Myasnikov, E.V. (2017) Fast techniques for nonlinear mapping of hyperspectral data. *Proceedings of 9th International Conference on Machine Vision*, Vol. 10341 (SPIE), pp. 103411D-1–103411D-6.
[26] Myasnikov, E. (2018) Embedding spatial context into spectral angle based nonlinear mapping for hyperspectral image analysis. *Proceedings of Computer Vision and Graphics*. LNCS, Vol. 11114, pp. 263–274.
[27] Myasnikov, E. (2018) Nonlinear dimensionality reduction of hyperspectral data using spectral correlation as a similarity measure. *Proceedings of Analysis of Images, Social Networks and Texts*. LNCS, Vol. 10716, pp. 237–244.
[28] Myasnikov, E. (2019) Nonlinear dimensionality reduction of hyperspectral data based on spectral information divergence preserving principle. *Journal of Physics: Conference Series*, 1368, 032030.
[29] Myasnikov, E. (2022) Data dimensionality reduction technique based on preservation of Hellinger divergence. *Proceedings of Analysis of Images, Social Networks and Texts*. LNCS, Vol. 13217, pp. 190–198.
[30] Myasnikov, E. (2022) A feature fusion technique for dimensionality reduction. *Pattern Recognition and Image Analysis*, 32, 607–610.
[31] Roweis, S.T. and Saul, L.K. (2000) Nonlinear dimensionality reduction by locally linear embedding. *Science*, 290, 2323–2326.
[32] Sammon Jr., J.W. (1969) A nonlinear mapping for data structure analysis. *IEEE Transactions on Computers*, C–18(5), 401–409.
[33] Shen-En, Q. and Guangyi, C. (2007) A new nonlinear dimensionality reduction method with application to hyperspectral image analysis. *Proceedings of IEEE International Geoscience and Remote Sensing Symposium*, pp. 270–273.
[34] Sun, W., *et al.* (2014) Nonlinear dimensionality reduction via the ENH-LTSA method for hyperspectral image classification, *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 7(2), 375–388.

[35] Sun, W., Liu, C., and Li, W. (2015) Hyperspectral imagery classification using the combination of improved Laplacian eigenmaps and improved k-nearest neighbor classifier. *Wuhan Daxue Xuebao (Xinxi Kexue Ban)/Geomatics and Information Science of Wuhan University*, 40(9), 1151–1156.

[36] Tenenbaum, J.B., de Silva, V., and Langford, J.C. (2000) A global geometric framework for nonlinear dimensionality reduction. *Science*, 290, 2319–2323.

[37] Tenenbaum, J.B. and de Silva, V. (2003) Global versus local methods in nonlinear dimensionality reduction. *Advances in Neural Information Processing Systems*, 15, 705–712.

[38] van der Maaten, L.J.P., Postma, E.O., and van den Herik, H.J. (2009) Dimensionality reduction: A comparative review. *Tilburg University Technical Report*, TiCCTR 2009-005.

[39] Wang, L., Shi, C., Diao, C., Ji, W., and Yin, D. (2016) A survey of methods incorporating spatial information in image classification and spectral unmixing. *International Journal of Remote Sensing*, 37(16), 3870–3910.

[40] Yan, L. and Niu, X. (2014) Spectral-angle-based Laplacian Eigenmaps for nonlinear dimensionality reduction of hyperspectral imagery. *Photogrammetric Engineering and Remote Sensing*, 80(9), 849–861.

[41] Yan, L. and Roy, D.P. (2015) Improved time series land cover classification by missing observation- adaptive nonlinear dimensionality reduction. *Remote Sensing of Environment*, 158, 478–491.

[42] Ziemann, A.K. and Theiler, J. (2016) Graph-based and statistical approaches for detecting spectrally variable target materials. *Proceedings of Algorithms and Technologies for Multispectral, Hyperspectral, and Ultraspectral Imagery XXII*, Vol. 9840 (SPIE), p. 98400T.

This page intentionally left blank

© 2025 World Scientific Publishing Company
https://doi.org/10.1142/9789811267215_0010

Chapter 10

Intellectual Software Support in Problems of Digital Registration of Random Point Structures

Alexandr L. Reznik* **and Alexandr A. Soloviev†**

Institute of Automation and Electrometry
of the Siberian Branch of the Russian Academy of Sciences,
Novosibirsk, the Russian Federation
**reznik@iae.nsk.su*
†*solowey@rambler.ru*

The main features of this chapter are original methods developed that help in solving complex fundamental problems arising in the study of random discrete fields and digital images. These methods are based on not numerical calculations but on analytical computer transformations.

The second distinctive feature of the methods described in this chapter is that with their help it is possible to implement the idea in pure form expressed by John von Neumann: the researcher faces a problem that he can not solve, so he uses a computer to do the labor-intensive calculations, which can lead him to "right" answer, and if he's lucky (with the computer-obtained solution), he finds a rigorous mathematical proof.

Another feature of this chapter is that mathematical proof of the calculated probability formulas and dependencies forced us to determine the explicit analytical form of the generalized Catalan numbers (that naturally extend the classical Catalan sequence known since Leonhard Euler's time and arising in many applications of probability theory and mathematical statistics).

The developed algorithms were successfully applied to solving problems related to digital error-free discrete-point image recording, the construction of asymptotically optimal decorrelating transformations of signals with different smoothness, and the development of time-optimal localization algorithms for pulsed objects (dots) generating extremely short delta pulses in random time.

Keywords: Software support, Poisson source, analytical transformations, multidimensional integral, Young tables, generalized Catalan numbers, ballot problem, point-pulse source, optimal search strategy, multipurpose search.

10.1 Introduction

The main feature of this chapter is the development of unique methods for solving difficult fundamental problems connected with the study of random point fields and digital images for the purpose of transferring labor-intensive analytical (not numerical) calculations to a computer. In this case, the computer is not a powerful "counter" but an "intellectual tool" for carrying-out equivalent mathematical calculations, which the researcher cannot accomplish manually because of their enormous volume. Currently, a large number of programming languages and software packages for symbolic operations are well known: analytical computing system "Reduce" (developed by computer scientist Anthony Hearn, USA), "Schoonchip" (created by Nobel laureate Martinus J. G. Veltman), "MuMATH", "Macsyma", "Maple V", "Mathematica", etc. In Russia (USSR), the authors of the first analytical manipulation software systems who successfully applied them for scientific research were academicians Kantorovich, Yanenko, Glushkov, Shirkov and his colleagues (1960–1970), and Pottosin [3, 13, 15, 16, 30]. The algorithms and programs created and used by them proved to be effective for finding solutions to algebraic-differential equations, for carrying out polynomial calculations, and for solving the problematic tasks of theoretical physics. Although all listed systems have proved their effectiveness and found wide popularity among researchers, they are not competitors to the software packages described in this chapter. The reason is the universality of all the above systems does not allow them to be equally effective in all

applications. In many cases, it is required to create problem-oriented libraries that take into account the algorithmic specificity of the problems being solved. The specialized software packages of analytical transformation described in this chapter (based on the multidimensional integration over convex domains in n-dimensional space) are just a typical example of this kind.

The second distinctive feature of the methods described in this chapter is that with their help it became possible to confirm J. von Neumann's idea in a pure form: when the researcher faces a problem he is unable to solve by himself, he uses a computer to deal with intensive calculations that help him to see "right" answer and then in a case of luck (i.e. knowing the "computer-prompted" solution) conducts a rigorous and constructive proof. In our case, we were able to establish and then prove mathematically several accurate analytical formulas describing the probability of error-free reading of random discrete point images using computer calculations. It would be impossible (or very difficult) without serious software support.

Another feature of this chapter is that rigorous mathematical proof of the obtained probabilistic dependencies required the extension of the classical Catalan numbers we proposed for the multidimensional case (including their explicit form for several specific applications). This generalization turned out to be a useful tool for solving many theoretical and applied combinatorial probability problems.

The analytical software manipulation systems described in the following were effectively applied to solve problems from different fields of informatics, specifically the research of readout of random discrete point images, the construction of asymptotically optimal decorrelating transformations for a signal with different smoothness degrees, and the development of time-optimal search algorithms of random pulse-point sources [7–10,20–26]. However, within the framework of this chapter, the discussion is limited by probabilistic problems arising during the readout of random point structures and digital registration of random pulse-point fields. All materials are selected and assembled in such a way that the reader who is not interested in specific methods and details of the software implementation of libraries performing analytical calculations can only skim

(or even skip) Section 10.2 altogether, which practically does not affect the understanding of the material outlined in Sections 10.3 and 10.4.

10.2 Analytical Computer Transformations in Problems of Random Point Fields Analysis

10.2.1 *Origin of the problem*

This section describes mathematical methods that are the foundation for constructing specialized algorithms and programs required for the study of random point structures. The main task considered here is to establish the reliability of error-free readout of random two-dimensional fields in those cases when its digital registration is carried out with the help of scanning aperture with several threshold levels. A random two-dimensional field is meant for the realization of a random discrete point distribution created by a Poisson source with λ intensity on the plane. This means that the probability of the event that exactly n random samples of the flow will fall into any subset A of such a field depends only on the square $S(\mathrm{A})$ of this subset but does not depend on any other of its space-topological characteristics and is given by the probability

$$P_A(n) = \exp(-\lambda s(A))\frac{(\lambda s(A))^n}{n!}. \tag{10.1}$$

The classical registering method of the coordinates of such point objects is successive "television" readout (with the aperture $\varepsilon \times \varepsilon$) of the horizontal ε-width bands of the image. Fixation of point object's coordinates occurs at the moment of its entry into the borders of the scanning aperture. The most important indicator of this procedure's reliability is the probability that no more than k objects will be located in the integrating aperture's window for the whole scanning period (here k is the number of threshold levels of the integrator aperture, and exceeding the threshold level means missing the object).

A general analytical solution for this problem cannot be obtained yet. Only partial formulas for the simplest statements of the problem

are known (for example, in [18, 31], the accurate analytic solution is given for a readout with a single-threshold-level integrator); in addition, a series of asymptotic relations were obtained in [6]. Sections 10.2.1 and 10.2.2 are related to the development of the methods and the construction of the algorithms for solving various modifications of this problem, but before moving on to the details, let us formulate a one-dimensional problem (in principle, the multithreshold readout problem is reduced to this task):

"Let n points $x_1, x_2, \ldots, x_n$ be randomly thrown on the interval $(0, L)$. What is the probability $P_{n,k}(\varepsilon, L)$ that there is no subinterval of length Ω_ε containing more than k points?"

An analytical solution to this problem is known [18, 31] only for the case when $k = 1$:

$$P_{n,1}(\varepsilon, L) = (L - (n-1)\varepsilon)^n / L^n, \quad (0 \le \varepsilon \le L/(n-1)). \tag{10.2}$$

Unfortunately, the existing methods for solving problems associated with the random partitioning of the interval (this problem also belongs to them) [2, 4, 5, 17] do not allow us to calculate the probabilities $P_{n,k}(\varepsilon, L)$ for any n and k, so we implemented some ideas based on analytical transformations on a computer.

10.2.2 *Estimation of reliability of error-free readout of random point fields via computer analytical multidimensional integral calculations*

This section is devoted only to the calculation of the formulas $P_{n,k}(\varepsilon, L)$ using programs of direct integration on parametrically defined polyhedra in n-dimensional space. The brief essence of algorithms is here. First, in the calculating scheme for the formulas $P_{n,k}(\varepsilon, L)$, the interval $(0, L)$ is replaced by the interval $(0, 1)$ with the help of normalization. Thus, the parameter L is excluded from further consideration. Then the probability $P_{n,k}(\varepsilon)$ (it's $P_{n,k}(\varepsilon/L, 1)$ in the previous notation) is represented in the form

$$P_{n,k}(\varepsilon) = n! \int\limits_{D_{n,k}(\varepsilon)} \ldots \int dx_1 \ldots dx_n, \tag{10.3}$$

where the integration domain $D_{n,k}(\varepsilon) \subset R^n$ is described by a system of linear inequalities

$$\begin{cases} 0 < x_1 < x_2 < \cdots < x_{n-1} < x_n < 1, \\ x_{k+1} - x_1 > \varepsilon, \\ x_{k+2} - x_2 > \varepsilon, \\ \qquad \vdots \\ x_n - x_{n-k} > \varepsilon. \end{cases} \tag{10.4}$$

In fact, the integration (10.3) on the area (10.4) means that inside of the unit n-dimensional cube $K_n = \{X = (x_1, x_2, \ldots, x_n)/0 < x_i < 1,\ i = 1, 2, \ldots, n\}$ we calculate the volume of the polyhedron $S_n = \{X = (x_1, x_2, \ldots, x_n)/x_{k+i} - x_i > \varepsilon,\ i = 1, 2, \ldots, n-k\}$. The ratio of the volume of the polyhedron S_n to the volume of the unit cube K_n is exactly the measure corresponding to the sought probability $P_{n,k}(\varepsilon)$. Further, the integral (10.3) on the area (10.4) is written in the equivalent form

$$\begin{aligned} P_{n,k}(\varepsilon) = n! \int_{-\infty}^{+\infty} \ldots \int_{-\infty}^{+\infty} \\ \times 1[x_1]1[x_2] \ldots 1[x_n - x_{n-1}]1[1 - x_n]1[x_{k+1} - x_1 - \varepsilon] \\ \times 1[x_{k+2} - x_2 - \varepsilon] \ldots 1[x_n - x_{n-k} - \varepsilon] dx_1 \ldots dx_n, \end{aligned} \tag{10.5}$$

so integration is already carried out on all space, and the indicator function of the integration domain $D_{n,k}(\varepsilon)$ is transformed into the product of the unit functions $1[z] = \{ {}^{0,\ z \le 0,}_{1,\ z > 0,}$ that form the integrand of the multiple integral (10.5). Then the n-dimensional integral (10.5) by successive application of relation

$$\left(\prod_{j=1}^{l} 1[x_r - \alpha_j] \right) \left(\prod_{i=1}^{m} 1[\beta_i - x_r] \right)$$

$$= \sum_{j=1}^{l} \sum_{i=1}^{m} 1[x_r - \alpha_j]1[\beta_i - x_r]$$

$$\times \left\{ 1[\beta_i - \alpha_j] \left(\prod_{\substack{q=1 \\ q \neq j}}^{l} 1[\alpha_j - \alpha_q] \right) \left(\prod_{\substack{s=1 \\ s \neq i}}^{m} 1[\beta_s - \beta_i] \right) \right\} \tag{10.6}$$

is reduced to a set of repeated integrals with already calculated integration limits. If we calculate the integration limits on variable x_1 in the initial integral (10.3) using the algorithmic scheme (10.6), then it is divided into two integrals:

$$\begin{aligned} P_{n,k}(\varepsilon) = n! \int_{-\infty}^{\infty} \dots \int_{-\infty}^{\infty} \left[\int_0^{x_2} dx_1 \right] \\ \times 1[x_{k+1} - x_2 - \varepsilon] 1[x_2] \left(\prod_{m=3}^{n} 1[x_m - x_{m-1}] \right) \\ \times 1[1 - x_n] \left(\prod_{m=2}^{n-k} 1[x_{m+k} - x_m - \varepsilon] \right) dx_2 \dots dx_n \\ + n! \int_{-\infty}^{\infty} \dots \int_{-\infty}^{\infty} \left[\int_0^{x_{k+1} - \varepsilon} dx_1 \right] 1[x_2 + \varepsilon - x_{k+1}] \\ \times 1[x_{k+1} - \varepsilon] \left(\prod_{m=3}^{n} 1[x_m - x_{m-1}] \right) \\ \times 1[1 - x_n] \left(\prod_{m=2}^{n-k} 1[x_{m+k} - x_m - \varepsilon] \right) dx_2 \dots dx_n. \end{aligned} \tag{10.7}$$

When we calculate integration limits according to relation (10.6) for each variable x_2, then x_3, and so on, as a result, we transform original multiple integral (10.3) to the sum of repeated integrals with already computed integration limits:

$$\begin{aligned} P_{n,k}(\varepsilon) = 1[\varepsilon - a_1] 1[b_1 - \varepsilon_1] \times \int_{\alpha_{n,1}(\varepsilon)}^{\beta_{n,1}(\varepsilon)} dx_n \int_{\alpha_{n-1,1}(x_n,\varepsilon)}^{\beta_{n-1,1}(x_n,\varepsilon)} dx_{n-1} \dots \\ \times \int_{\alpha_{21}(x_3,\dots,x_n,\varepsilon)}^{\beta_{21}(x_3,\dots,x_n,\varepsilon)} dx_2 \int_{\alpha_{11}(x_2,\dots,x_n,\varepsilon)}^{\beta_{11}(x_2,\dots,x_n,\varepsilon)} dx_1 \end{aligned}$$

$$
\begin{aligned}
&+1\left[\varepsilon-a_2\right]1\left[b_2-\varepsilon_1\right]\times\int_{\alpha_{n,2}(\varepsilon)}^{\beta_{n,2}(\varepsilon)}dx_n\int_{\alpha_{n-1,2}(x_n,\varepsilon)}^{\beta_{n-1,2}(x_n,\varepsilon)}dx_{n-1}\ldots\\
&\times\int_{\alpha_{22}(x_3,\ldots,x_n,\varepsilon)}^{\beta_{22}(x_3,\ldots,x_n,\varepsilon)}dx_2\int_{\alpha_{12}(x_2,\ldots,x_n,\varepsilon)}^{\beta_{12}(x_2,\ldots,x_n,\varepsilon)}dx_1\\
&+\cdots\\
&+1\left[\varepsilon-a_N\right]1\left[b_N-\varepsilon_1\right]\\
&\times\int_{\alpha_{n,N}(\varepsilon)}^{\beta_{n,N}(\varepsilon)}dx_n\int_{\alpha_{n-1,N}(x_n,\varepsilon)}^{\beta_{n-1,N}(x_n,\varepsilon)}dx_{n-1}\ldots\\
&\times\int_{\alpha_{2N}(x_3,\ldots,x_n,\varepsilon)}^{\beta_{2N}(x_3,\ldots,x_n,\varepsilon)}dx_2\int_{\alpha_{1N}(x_2,\ldots,x_n,\varepsilon)}^{\beta_{1N}(x_2,\ldots,x_n,\varepsilon)}dx_1. \qquad (10.8)
\end{aligned}
$$

At the final stage, first, we successively integrate each of the N repeated integrals into which the original integral (10.3) is converted, and then the results are combined according to free parameter ε boundaries. This algorithm allows us to constructively calculate the formulas $P_{n,k}(\varepsilon)$, and it completely solves the problem of finding the probability $P_{n,k}(\varepsilon)$ for specified values n and k. The main defect of this method is that for $n > 3$ we cannot establish the limits of integration, check all the intermediate inequality's systems for compatibility and directly calculate the repeated integrals manually (it is really impossible because of extra large amount of necessary calculations). Therefore, we created the software package based on analytical transformations in full accordance with the scheme described above. Using them, formulas $P_{n,k}(\varepsilon)$, presented in Table A.1 (see Appendix), were calculated.

10.2.3 *Recursive analytical computer programs in two-threshold problem of random point images readout*

In this paragraph, we describe a more rapid (in comparison with the previously shown) highly specialized algorithm for calculating the probabilities $P_{n,k}(\varepsilon)$. The term "two-threshold readout" means that the process of registering the coordinates of random point objects

is performed by an integrator having two threshold levels. Thus, in the following, we discuss methods and algorithms for calculating the formulas $P_{n,k}(\varepsilon)$ for $k = 2$. The basis of the new scheme is the creating of recursive equations for the discrete-combinatorial model of random point image recording process. The necessity to develop a new method is explained by the fact that the complexity of the algorithms described in Section 10.2.2 did not allow us to calculate the formulas $P_{n,k}(\varepsilon)$ for values of $n \geq 9$ (although, with parallel programming methods, their speed was significantly improved but not enough to establish new general analytical dependencies).

Thus, by analogy with the known solution (10.2) for $k = 1$, we tried to find the general analytic solution $P_{n,k}(\varepsilon)$ for the case $k = 2$. In contrast to the multidimensional integration algorithms described above, a fundamentally different mathematical approach was used to create the new scheme. In the first stage, recursive relations were constructed using purely combinatorial methods and then with their help discrete prototypes of the formulas $P_{n,2}(\varepsilon)$ were calculated. Then, for successively increasing values of the integer parameter n, these discrete analog prototypes were used to find the "continuous" probability formulas $P_{n,2}(\varepsilon)$ (by passage to the limit). The brief idea of the combinatorial model is here. The interval $(0, L)$ was interpreted as a set of r equal discretes. The random throwing of n points on the interval $(0, L)$ was replaced by a random throwing of n indistinguishable balls on r urns. The unit of l adjacent discrete was an analog of the subinterval with ε length. If none of these l-subintervals from the initial r-interval $(0, L)$, had more than 2 points, the throwing was accepted as "successful", and the ratio of the total number of "successful" throwings $Q(r, n, l)$ to the total outcome number of the experiment $Q(r, n)$ was accepted as an integer analog of the probability $P_{n,2}(\varepsilon)$. If it were possible to find a closed analytic expression for the total number of "successful" throwings $Q(r, n, l)$, this would be equivalent to the complete solution of the problem for $k = 2$. In this case, the probability $P_{n,2}(\varepsilon)$ would be calculated by a simple passage of the limit

$$P_{n,2}(\varepsilon) = \lim_{\substack{r\to\infty \\ (l/r)\to\varepsilon}} \frac{Q(r, n, l)}{Q(r, n)} \tag{10.9}$$

since the expression for the total number of throwings $Q(r,n)$ is known [11]:

$$Q(r,n) = C_{n+r-1}^{r-1} = \frac{(n+r-1)!}{n!(r-1)!}. \tag{10.10}$$

Before we introduce the recursive relations used to calculate the number of "successful" throwings $Q(r,n,l)$, we set a number of agreements for further notation. The number of balls n can be written in the following form: $n = 2N + s$; $N = 0,1,2,\ldots$; $s = 1,2$. The total number of "successful" throwings of n balls on r urns in such a way that one ball got into the first urn, the second ball into the ith urn, and the rest of the balls got into urns whose current numbers are greater than $i (i \le l)$, we sign as $q(i,r,n,l)$.

For the fixed n, the functions $q(i,r,n,l)$ and $Q(r,n,l)$ are piecewise polynomial expressions relative to the remaining arguments. For polynomials describing the behavior of these functions in different areas, we use the following notation:

$$\begin{aligned}
q_0(i,r,n,l) &\equiv q(i,r,n,l) \quad \text{for } (N+2)l < r,\\
q_1(i,r,n,l) &\equiv q(i,r,n,l) \quad \text{for } (N+1)l < r \le (N+2)l,\\
q_2(i,r,n,l) &\equiv q(i,r,n,l) \quad \text{for } Nl < r \le (N+1)l,\\
Q_0(r,n,l) &\equiv Q(r,n,l) \quad \text{for } (N+1)l < r,\\
Q_1(r,n,l) &\equiv Q(r,n,l) \quad \text{for } Nl < r \le (N+1)l.
\end{aligned}$$

Then the recurring relations are the following (for $s = 2$ (n is even)):

$$\begin{aligned}
q_0(i,r,n,l) &= Q_0(r-l-i, n-2, l) + \sum_{i_1=0}^{i} Q_0(r-2l-i_1, n-3, l)\\
&\quad + \sum_{i_1=0}^{i} \sum_{i_2=i-i_1}^{l} q_0(i_2, r-l-i_1, n-2, l);\\
q_1(i,r,n,l) &= Q_0(r-l-i, n-2, l)\\
&\quad + \sum_{i_1=0}^{i} \sum_{i_2=i-i_1}^{r-Nl-i_1} q_0(i_2, r-l-i_1, n-2, l);
\end{aligned}$$

$$q_2(i,r,n,l) = 0;$$

$$Q_0(r,n,l) = \sum_{k=Nl}^{r} \sum_{i=0}^{k-Nl} q_1(i,k,n,l);$$

$$Q_1(r,n,l) = Q_1(r-1,n,l) + Q_1(r-l,n-1,l) + \sum_{i=0}^{l} q_0(i,r,n,l); \tag{10.11}$$

for $s = 1$ (n is odd):

$$q_0(i,r,n,l) = Q_0(r-l-i,n-2,l) + \sum_{i_1=0}^{i} Q_0(r-2l-i_1,n-3,l) + \sum_{i_1=0}^{i} \sum_{i_2=i-i_1}^{l} q_0(i_2,r-l-i_1,n-2,l);$$

$$q_1(i,r,n,l) = Q_1(r-l-i,n-2,l) + \sum_{i_1=0}^{i} Q_1(r-2l-i_1,n-3,l) + \sum_{i_1=0}^{i} \sum_{i_2=i-i_1}^{r-Nl-i_1} q_1(i_2,r-l-i_1,n-2,l) + \sum_{i_1=0}^{i} \sum_{i_2=r-Nl-i_1}^{l} q_2(i_2,r-l-i_1,n-2,l);$$

$$q_2(i,r,n,l) = \sum_{i_1=0}^{r-Nl} Q_1(r-2l-i_1,n-3,l) + \sum_{i_1=0}^{r-Nl} \sum_{i_2=i-i_1}^{l} q_2(i_2,r-l-i_1,n-2,l);$$

$$Q_0(r,n,l) = \sum_{k=Nl}^{r} \sum_{i=0}^{k-Nl} q_1(i,k,n,l);$$

$$Q_1(r,n,l) = \sum_{k=Nl}^{r} \left[Q_1(k-l, n-1, l) + \sum_{i=0}^{k-Nl} q_1(i,k,n,l) + \sum_{i=k-Nl}^{l} q_2(i,k,n,l) \right]. \tag{10.12}$$

According to these relations, for $n = 3, 4, 5, \ldots$, the formulas $Q_0(r,n,l)$ and $Q_1(r,n,l)$ as a function of r and l are calculated. Then passage to the limit is performed

$$\lim_{\substack{r\to\infty \\ l/r\to\varepsilon}} \left[\frac{Q_0(r,N,l)}{Q(r,n)} \right] \quad \text{and} \quad \lim_{\substack{r\to\infty \\ l/r\to\varepsilon}} \left[\frac{Q_1(r,N,l)}{Q(r,n)} \right]. \tag{10.13}$$

The initial data for the initiation of a recurrent-recursive computational process are taken from trivial considerations when $n = 0, 1, 2$. Because of huge amount of computations necessary to implement the described recursion, the whole algorithm of combinatorial calculation was again realized as a computer program. In this way, we managed to move forward in the calculation of the probabilities $P_{n,2}(\varepsilon)$ and obtain new previously unknown formulas. Moreover, the analysis of these "computer" formulas allowed us to see a general pattern and to put forward such a hypothesis: in the presence of two threshold levels (i.e. if $k = 2$), then for even n on the interval $1/(n/2) < \varepsilon < 1/((n/2) - 1)$, the following formula is correct:

$$P_{n,2}(\varepsilon) = (2/n) C_n^{(n/2)-1} (1 - ((n/2) - 1)\varepsilon)^n. \tag{10.14}$$

It should be noted that the formula (10.14) was "prompted" by a computer and published in 1981 [20], but only recently it was rigorously proved in 2011 [27]. This proof was the final stage in the realization of the idea of John von Neumann:

- At the first stage, the computer has calculated maximally possible number of partial formulas $P_{n,2}(\varepsilon)$ for concrete values of n.
- At the second stage as a result of the analysis of these program-calculated formulas, the assumption was made that for even n on

the interval $1/(n/2) < \varepsilon < 1/((n/2)-1)$ the probability $P_{n,2}(\varepsilon)$ is represented by the general formula (10.14).

- At the third final stage, the rigorous mathematical proof of "computer-prompted" answer (formula) was made (note that the integer coefficients in the "continuous" formula (10.14) are the classical Catalan numbers).

10.2.4 *Software calculation of probabilistic formulas by means of multidimensional integrals differentiation with respect to parameter*

In Section 10.1.3, we present a multidimensional integral (10.5) that needs to be calculated to find concrete probability formulas $P_{n,k}(\varepsilon)$. In several cases (for this purpose, the integration area (10.4) must satisfy certain specific constraints), to achieve a higher efficiency, the direct integration method described above can be modified if we take some following details into consideration. First, all the coefficients of the variables x_i in the Heaviside Functions that form the integrand of the expression (10.5) are equal ± 1, and this circumstance is directly used in the improved algorithm (that, strictly speaking, lies at its base). Second, the new modified method is oriented to finding the formulas $P_{n,k}(\varepsilon)$ only inside of the range of variation of parameter ε that contains the point $\varepsilon = 0$. The algorithm is based on the fact that any n-degree polynom $P(\varepsilon)$ can be uniquely reconstructed by its value and the values of all its derivatives of the parameter ε at some definite point (in our case this point is $\varepsilon = 0$):

$$P(\varepsilon) = \sum_{i=0}^{n} \frac{1}{i!} \frac{d^i P}{d\varepsilon^i}(0)\varepsilon^i. \tag{10.15}$$

To calculate the first and the next derivatives, we use the fact that in this case the operations of integration and differentiation are commutative:

$$\frac{dP(\varepsilon)}{d\varepsilon} = n! \int \cdots \int \frac{d}{d\varepsilon}\{1[\cdots] \times \cdots \times 1[\cdots]\} dx_1 \ldots dx_n. \tag{10.16}$$

Then, the following simple relations are required:

$$\frac{d}{dz}1[z] = \delta(z); \tag{10.17}$$

$$\int_{-\infty}^{+\infty} \delta(z)F(z)dz = F(0). \tag{10.18}$$

Here $\delta(z)$ is the Dirac delta function.

The differentiation under integral in the expression (10.16) can be performed according to standard rules of differentiation of function's multiplication taking into account relation (10.17). So the integrand transfers into a sum of multiplications whose factors are the set of Heaviside Functions plus one delta function as a cofactor. The resulting integral (after the differentiation) may be presented as the sum of integrals — their number is the same as the number of summands in integrand expression. If now we concretize integral (10.16) by means of expression (10.5), the result is

$$\begin{aligned}
\frac{dP(\varepsilon)}{d\varepsilon} = n! \int \cdots \int & 1[x_1] \times 1[x_2 - x_1] \times \cdots \times 1[x_n - x_{n-1}] \\
& \times 1[1 - x_n] \times \delta[x_{k+1} - x_1 - \varepsilon] \times 1[x_{k+2} - x_2 - \varepsilon] \\
& \times \cdots \times 1[x_n - x_{n-k} - \varepsilon] dx_1 \dots dx_n \\
+ n! \int \cdots \int & 1[x_1] \times 1[x_2 - x_1] \times \cdots \times 1[x_n - x_{n-1}] \\
& \times 1[1 - x_n] \times 1[x_{k+1} - x_1 - \varepsilon] \times \delta[x_{k+2} - x_2 - \varepsilon] \\
& \times \cdots \times 1[x_n - x_{n-k} - \varepsilon] dx_1 \dots dx_n \\
& \vdots \\
+ n! \int \cdots \int & 1[x_1] \times 1[x_2 - x_1] \times \cdots \times 1[x_n - x_{n-1}] \\
& \times 1[1 - x_n] \times 1[x_{k+1} - x_1 - \varepsilon] \times 1[x_{k+2} - x_2 - \varepsilon] \\
& \times \cdots \times \delta[x_n - x_{n-k} - \varepsilon] dx_1 \dots dx_n.
\end{aligned} \tag{10.19}$$

In expression (10.19), each of the summands can be simplified by integrating one of the variables entering into the argument of the corresponding delta function. In fact, it is only necessary, using the relation (10.18), to make the corresponding substitution into the

arguments of either of the Heaviside functions. As a result of such sequential transformations, the expression $\frac{dP(\varepsilon)}{d\varepsilon}$ takes the form of the sum of integrals that have a multiplicity of one unit less than the original integral (10.5). The general form of the integrands — the multiplication of Heaviside functions — remains invariable. Therefore, it is possible to reuse the described procedure to calculate the highest derivatives without any changes in the structural scheme.

So, any derivative $\frac{d^j P(\varepsilon)}{d\varepsilon^j}$ $(j = 0, 1, \ldots, n)$ is represented as a sum of integrals of $(n - j)$th multiplicity. Its value at zero is calculated as follows. We substitute the value $\varepsilon = 0$ to any of the integrals of $(n-j)$th multiplicity. Obviously, the set of Heaviside functions under integral contains trivial information: either the system is inconsistent (so, the integral is zero), or this system imposes restrictions only on the variable's order within the interval $(0, 1)$. In the latter case, the integral is equal to $1/(n - j)!$, where j is the order of the derivative whose numerical value is determined at the point $\varepsilon = 0$. Counting the number of non-zero integrals, we get the concrete value of the first derivative $\frac{dP(\varepsilon)}{d\varepsilon}\Big|_{\varepsilon=0}$, then $\frac{d^2P(\varepsilon)}{d\varepsilon^2}\Big|_{\varepsilon=0}$, etc. After all derivatives' values at zero are determined by the described scheme, the coefficients of the unknown polynomial are found from the relation (10.15). Note it is sufficient to use only integer arithmetic in the described algorithmic scheme.

On the computer, the process of solving the described problem is a consequent implementation of the above algorithm. Calculated by this method the specific formulas $P_{n,k}(\varepsilon)$, describing the probability of error-free registration of n-point image when integrating aperture has k threshold levels were included in the composite Table A.1 (see Appendix). Analyzing the results obtained, we succeeded in establishing a formula for the probability $P_{n,2}(\varepsilon)$ that is correct for any n inside of the range $0 < \varepsilon < 1/(\text{entier } (n + 1)/2)$:

$$\begin{aligned} P_{n,2}(\varepsilon) = {} & C_n^0 + C_n^2(-n+2)\varepsilon^2 + C_n^3(4n-10)\varepsilon^3 \\ & + C_n^4(3n^2 - 37n + 86)\varepsilon^4 + C_n^5(-40n^2 + 394n - 922)\varepsilon^5 \\ & + C_n^6(-15n^3 + 625n^2 - 5171n - 12086)\varepsilon^6 \\ & + C_n^7(420n^3 - 10724n^2 + 79996n - 187002)\varepsilon^7 \\ & + C_n^8(105n^4 - 10570n^3 + 205499n^2 - 1426841n \end{aligned}$$

$$+3336406)\varepsilon^8 + C_n^9(5040n^4 - 155708n^3 + 2267664n^2 - 17317506n + 52315558)\varepsilon^9 + C_n^{10}(-945n^5 + 189000n^4 - 15794625n^3 + 389687181n^2 - 3798029823n + 12998966646)\varepsilon^{10} + o(\varepsilon^{10}). \tag{10.20}$$

In principle, the relation (10.20) is a well-calculated "computer" hypothesis. Although all the computer calculations confirm it, the established dependence needs rigorous mathematical proof.

10.2.5 *A perspective computer-based method of multidimensional analytical integration*

The high speed of the algorithms described in Sections 10.2.1–10.2.4 and the programs created on their basis have allowed us to make significant progress in calculating the formulas $P_{n,k}(\varepsilon)$, but the overall analytically closed solution of the original problem has yet to be found. To advance further in this problem, we have proposed one more original algorithmic approach to this problem. Its essence is to use the direct integration immediately in the general analytical form and quite so to find formulas successively for the probabilities $P_{n,n-1}(\varepsilon), P_{n,n-2}(\varepsilon), P_{n,n-3}(\varepsilon)$, etc. In contrast to all previous algorithms, in this scheme, there are already two variables free: not only a continuous parameter ε but also an integer parameter n. Let's demonstrate a new computational procedure on the simplest example: calculation of the probability $P_{n,n-1}(\varepsilon)$. The expression

$$P_{n,n-1}(\varepsilon) = n! \int\limits_{D_{n,n-1}(\varepsilon)} \cdots \int dx_1 \dots dx_n, \tag{10.21}$$

$$D_{n,n-1}(\varepsilon) = \begin{cases} 0 < x_1 < x_2 < \cdots < x_{n-1} < x_n < 1 \\ x_n - x_1 > \varepsilon \end{cases} \tag{10.22}$$

is converted to the equivalent form

$$P_{n,n-1}(\varepsilon) = n! \int_\varepsilon^1 dx_n \int_0^{x_n-\varepsilon} dx_1 \left[\int_{x_1}^{x_n} dx_{n-1} \dots \int_{x_1}^{x_4} dx_3 \int_{x_1}^{x_3} dx_2 \right]. \tag{10.23}$$

After the direct integration, we get the final formula

$$\begin{aligned} P_{n,n-1}(\varepsilon) &= n! \int_{\varepsilon}^{1} dx_n \int_{0}^{x_n-\varepsilon} dx_1 \left[\frac{(x_n - x_1)^{n-2}}{(n-2)!} dx_1 \right] \\ &= n! \int_{\varepsilon}^{1} \left(-\frac{\varepsilon^{n-1}}{(n-1)!} + \frac{x_n^{n-1}}{(n-1)!} \right) dx_n \\ &= n! \left(-\frac{\varepsilon^{n-1}(1-\varepsilon)}{(n-1)!} + \left(\frac{1}{n!} - \frac{\varepsilon^n}{n!} \right) \right) \\ &= 1 - \varepsilon^n - n\varepsilon^{n-1}(1-\varepsilon). \end{aligned} \tag{10.24}$$

Algorithm for calculating the probability

$$P_{n,n-2}(\varepsilon) = n! \int \cdots \int_{D_{n,n-2}(\varepsilon)} dx_1 \dots dx_n \tag{10.25}$$

is more complicated, since a multiple integral (10.25) cannot be converted to a single repeated integral of the type (10.24), but it is possible, using (10.6), to represent it as a sum of integrals. Omitting all the intermediate calculations, we give only the result obtained:

$$P_{n,n-2}(\varepsilon) = \begin{cases} 1 - 2C_n^2 \varepsilon^{n-2}(1-\varepsilon)^2 - 2\varepsilon^n, & 0 \le \varepsilon \le (1/2); \\ 1 - 2\varepsilon^n + (2\varepsilon - 1)^n - 2C_n^2 \varepsilon^{n-2}(1-\varepsilon)^2, & (1/2) \le \varepsilon \le 1. \end{cases} \tag{10.26}$$

The procedure for calculating the probability $P_{n,n-3}(\varepsilon)$ is even more unwieldy, so we give only the final result, omitting all the intermediate calculations:

$$P_{\substack{n,n-3 \\ (n>6)}}(\varepsilon) = \begin{cases} 1 - 2\varepsilon^n + C_n^1(6\varepsilon^n - 4\varepsilon^{n-1}) + C_n^2(-3\varepsilon^n + \varepsilon^{n-2}) \\ \quad + C_n^3(9\varepsilon^n - 18\varepsilon^{n-1} + 12\varepsilon^{n-2} - 3\varepsilon^{n-3}), \\ \qquad 0 \le \varepsilon \le (1/2); \\ 1 - 2\varepsilon^n + (2\varepsilon - 1)^n + C_n^1(1-\varepsilon)(-2\varepsilon^{n-1} + 2(2\varepsilon - 1)^{n-1}) \\ \quad + C_n^2(1-\varepsilon)^2(\varepsilon^{n-2} + (2\varepsilon - 1)^{n-2}) - 3C_n^3 \varepsilon^{n-3}(1-\varepsilon)^3, \\ \qquad (1/2) \le \varepsilon \le 1. \end{cases} \tag{10.27}$$

In contrast to all the previously given relations, formulas (10.24), (10.26), and (10.27) are manually calculated. The formalization of

the algorithm for obtaining by the computer all the analytic relations $P_{n,n-i}(\varepsilon)$ for any values of the integer parameter i is very complicated (since the calculations need to be programmed with two free parameters: n and ε), but possible procedure. The existence of an opportunity to realize the latest algorithm as a software product allows us to look optimistically at the prospect of progress in the task of finding analytical relationships for the probability of error-free readout of random point images, including finding the absolutely closed solution.

When researches use programs of analytical transformation, an important issue is the verification of the formulas and relationships obtained. In our case, the correctness of the obtained results is confirmed by the following circumstances: first, the mathematical validity of all the computational schemes for performed calculations; second, the solutions are the same for calculated analytic relations and known classical formulas (for example, for single-threshold readout); third, the identity of the results calculated by different computational schemes.

The results presented in Sections 10.2.1–10.2.5 were published in [7–10, 20, 21, 25, 26].

10.3 Proof of "Computer" Formulas with the Aid of Generalized Catalan Numbers

10.3.1 *The proof of the analytic relations using computer-calculated formulas and the classical Catalan sequence*

The main task of this section is a strict mathematical proof of the probability formulas $P_{n,k}(\varepsilon)$ for $k = 2$. It was noted in Section 10.2 that for $k = 1$ the dependence $P_{n,k}(\varepsilon)$ can be represented not only as an unwieldy multiple integral (10.3) on the domain (10.4) but also in the form of a compact repeated integral

$$P_{n,1}(\varepsilon) = n! \int_{(n-1)\varepsilon}^{1} dx_n \left\{ \int_{(n-2)\varepsilon}^{x_n-\varepsilon} dx_{n-1} \cdots \right.$$
$$\left. \times \left\{ \int_{2\varepsilon}^{x_4-\varepsilon} dx_3 \left\{ \int_{\varepsilon}^{x_3-\varepsilon} dx_2 \left\{ \int_{0}^{x_2-\varepsilon} dx_1 \right\} \right\} \right\} \right\}. \quad (10.28)$$

By successive integration of the expression (10.28) on variables $x_1, x_2, \ldots, x_n$, it is easy to obtain the earlier given formula (10.2). Unfortunately, for $k > 1$, the integral (10.3) on the domain (10.4) does not transform into a single repeated integral with already set integration limits, so the general analytical solution of the problem cannot be found not only for any value k, but difficulties arise even for $k = 2$. Therefore, to calculate the formulas $P_{n,2}(\varepsilon)$ corresponding to the two-threshold readout in addition to the methods of multidimensional integration, we have developed [20] the earlier mentioned special combinatorial-recursive model (in this scheme the continuous process of random point objects readout is replaced by its discrete analog with random throwing balls into urns). The combinatorial-recurrent scheme obtained by means of this model was completely formalized and programmed as well as the earlier scheme of multidimensional integration. The application of the created computer analytical programs allowed us not only to make significant progress in the calculation of the partial formulas $P_{n,2}(\varepsilon)$ but also to find new previously unknown dependencies. In particular, the analysis of computer results has allowed us to formulate the hypothesis [9] that for even values n, the earlier given formula (10.14) is correct (parameter ε change range is $1/(n/2) < \varepsilon < 1/((n/2)-1)$). The first (truth, very bulky) proof of this relation was presented by us in [27]. Following is a more compact proof using the classical Catalan numbers.

Proof of the "Computer" Hypothesis: So, it is required to prove that for the random throwing of $n = 2m$ points $x_1, x_2, \ldots, x_{2m}$ onto the interval $(0, 1)$, the probability that there will be no subinterval Ω_ε with length ε with more than 2 points is

$$P = (C_{2m}^{m} - C_{2m}^{m-1})(1 - (m-1)\varepsilon)^{2m}, \tag{10.29}$$

if $\frac{1}{m} < \varepsilon < \frac{1}{m-1}$.

We note that the expression (10.29) is equivalent to the formula hypothesis (10.14) and differs from it only by the form of the first cofactor, which does not depend on the parameter ε and it's the mth element of the classical Catalan sequence. In the following, we formulate a series of statements; formula (10.29) is the direct result of them.

Statement 1. Any placement of $2m$ random points within the interval $(0, 1)$ satisfying the conditions of the problem has to satisfy the

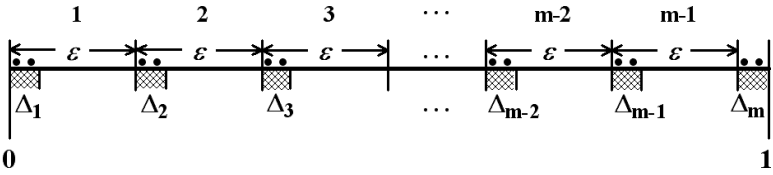


Figure 10.1. The scheme for placing an even ($n = 2m$) number of random points on the interval $(0, 1)$, when there is no subinterval $\Omega_\varepsilon \subset (0, 1)$ containing more than 2 points. The case for $1/m < \varepsilon < 1/(m-1)$.

scheme shown in Figure 10.1. In other words, all the hatched subintervals Δ (all of them have equal length $\Delta = 1-(m-1)\varepsilon$) has to contain 2 points, and "gaps" between hatched subintervals must be empty. To confirm this statement, there is enough to transform the inequality $1/m < \varepsilon < 1/(m-1)$ into an equivalent form $(m-1)\varepsilon < 1 < m\varepsilon$.

Statement 2. The probability P_1 that $n = 2m$ points thrown randomly into the interval $(0, 1)$ are placed according to the scheme in Figure 10.1 is

$$P_1 = [(m(1-(m-1)\varepsilon)^{2m}] \times \left[\frac{(2m)!}{m^{2m}2^m}\right]$$
$$= (1-(m-1)\varepsilon)^{2m}\frac{(2m)!}{2^m}. \tag{10.30}$$

In this expression, the first cofactor in square brackets corresponds to the probability that all $2m$ points will get into a common zone $\cup_{i=1}^{i=m}\Delta_{\rm i}$ uniting m hatched subintervals Δ_i, and the second cofactor corresponds to the probability that there are exactly two points in each hatched subinterval (this probability is easy to prove by induction).

Statement 3. If we follow the scheme of the diagram (Figure 10.1), the probability that in the variational series $\{0 < x_{i_1} < x_{i_2} < \cdots < x_{i_{2m}} < 1\}$ the minimal distance between the nearest odd points and the minimal distance between the nearest even points exceed ε (this condition is equivalent to the absence of inner ε-subintervals Ω_ε contained more than two points) is

$$P_2 = 2^m\frac{C_{2m}^m - C_{2m}^{m-1}}{(2m)!}. \tag{10.31}$$

Table 10.1. First kind.

ξ_1	ξ_3	$\cdots$	$\cdots$	$\cdots$	ξ_{2m-1}
ξ_2	ξ_4	$\cdots$	$\cdots$	$\cdots$	ξ_{2m}

Table 10.2. Second kind.

$\min(\xi_1, \xi_2)$	$\min(\xi_3, \xi_4)$	$\cdots$	$\cdots$	$\cdots$	$\min(\xi_{2m-1}, \xi_{2m})$
$\max(\xi_1, \xi_2)$	$\max(\xi_3, \xi_4)$	$\cdots$	$\cdots$	$\cdots$	$\max(\xi_{2m-1}, \xi_{2m})$

To verify the validity of the formula (10.31), we compare two experiments:

(1) In the first experiment, $n = 2m$ independent observations of a random variable ξ are carried out. The observed values $\xi_1, \xi_2, \ldots, \xi_{2m}$ are recorded in a table of the first kind (with dimensions of $2 \times m$, see Table 10.1) formed in the following order:
(2) In the second experiment, the $2 \times m$-table (table of the second kind, see Table 10.2) is formed as follows:

The following question arises: "What is the probability that the given tables will be Young tables (i.e., that their elements increase in each row and in each column)?"

For the first experiment, the sought probability is

$$P = \frac{C_{2m}^{m} - C_{2m}^{m-1}}{(2m)!}. \tag{10.32}$$

Here the numerator corresponds to the number of different Young tables of $2 \times m$ size (it is known [12] that their number is equal to mth element of the classical Catalan sequence), and the denominator corresponds to the total number of permutations of $2m$ elements.

Concerning the second experiment, it should be noted that if the output table of the second kind is a Young table, then a set of 2^m tables of the first kind leads to it (can be obtained by element's permutation in one or several columns of the first kind table). Therefore, to find the correct answer for the second table, probability (10.32) must be multiplied by 2^m. So, we have proved the formula (10.31).

Now it is necessary to be convinced that the scheme in Figure 10.1 corresponds in accuracy to experiment no.2. For this, it is sufficient to introduce variables $\tilde{x}_{i_j} = x_{i_j} - \varepsilon \times entier\left(\frac{j-1}{2}\right)$, $j = \overline{1, 2m}$, i.e. to replace the coordinates of random points from the variational series $\{x_{i_j}\}$ by their coordinates taken "on the module ε". Obviously, if we follow the diagram of Figure 10.1, the nearest even and nearest odd points are remote from each other on distance exceeding ε if and only if both sequences increase:

$$\{y_1 < y_2 < \cdots y_m\} \quad \text{and} \quad \{z_1 < z_2 < \cdots z_m\},$$

where $y_j = \min(\tilde{x}_{i_{2j-1}}, \tilde{x}_{i_{2j}})$, $j = \overline{1, m}$; $z_j = \max(\tilde{x}_{i_{2j-1}}, \tilde{x}_{i_{2j}})$, $j = \overline{1, m}$. So, the second experiment corresponds in accuracy to the scheme shown in Figure 10.1.

Completion of the Proof: Formula (10.29) is valid only if statements 1 and 2 are simultaneously satisfied. Multiplying the probabilities (10.30) and (10.31), we get formula (10.29), which confirms the validity of the hypothesis-formula (10.14).

10.3.2 *Generalized Catalan numbers and their use in the proof of analytic formulas*

In Section 10.2.1, the "computer-prompted" probability formula $P_{n,2}(\varepsilon)$ was strictly proved. This formula (10.29) corresponds to the reliability of correct readout of a random n-point image by an integrator with two threshold levels. The proved formula (10.29) is valid if the value of the integer parameter n is even ($n = 2m$) and continuous parameter ε belongs to subrange $1/m < \varepsilon < 1/(m-1)$.

Finding an analogous analytic dependence for odd values n ($n = 2m + 1$) is the subject of the nearest (within paragraph 1.2.2) consideration. To establish and to prove it has appeared not only a much more serious problem but required special research method development. In particular, at one proof's step, we had to introduce the concept and find explicit expressions for generalized Catalan numbers that are a natural three-dimensional extension of the classical one-dimensional Catalan sequence that appears in many applications of probability theory and mathematical statistics.

Problem Definition: So, the problem solved in the following is formulated as follows:

"Let $n = 2m + 1$ points be randomly thrown into the interval $(0, 1)$. It is required to find the probability $P_{2m+1,2}(\varepsilon)$ of the event that inside of the interval $(0, 1)$ there is no subinterval Ω_ε of length ε containing more than 2 points, if $1/(m + 1) < \varepsilon < 1/m$".

Problem Solution: The sequential solution of this problem given in the following is constructed in such a way that at first the main "statements" are given briefly (without proof), further (directly after them) the solution of the problem follows, and then necessary explanations and calculations are given. Let us immediately notice that "statements" are unequal in terms of complexity and "scientific" contribution. For example, "statement 1" is an ordinary combinatorial relation, "statements 2, 5, and 6" belong to the same type and are fairly simple, but "statement 7" is a complicated problem, requiring its solution to use generalized three-dimensional Catalan numbers, to find their explicit form and to construct special probability-geometric scheme. The proof of the "final statement" requires a good technique for combinatorial formula handling.

The initial statement: Any placement of random points $x_1 < x_2 < \cdots < x_{2m+1}$ on the interval $(0, 1)$ satisfying requirements of the problem formulated above, corresponds to the scheme shown in Figure 10.2.

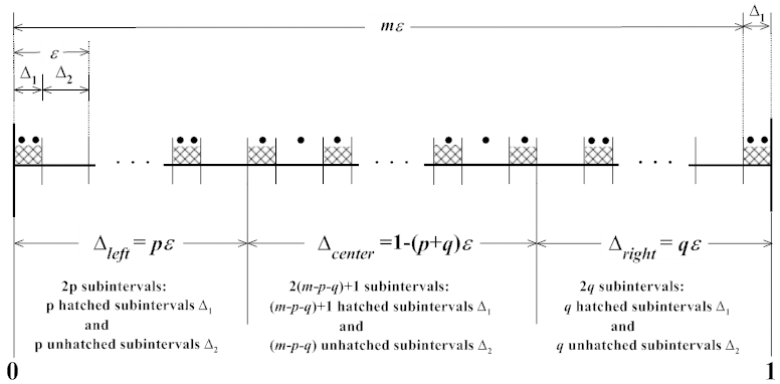


Figure 10.2. The placement of an odd $n = 2m + 1$ number of points on the interval $(0, 1)$, when there is no subinterval $\Omega_\varepsilon \subset (0, 1)$ of length ε containing more than 2 points. Case for $1/(m + 1) < \varepsilon < 1/m$.

In other words, the above statement means the following: if, in the case of a random throwing of an odd number $n = 2m + 1$ points on the interval $(0, 1)$, they are distributed in such a way that there are no more than two points in any inner ε-subinterval, then their placement on the interval $(0, 1)$ has to correspond to the scheme shown in Figure 10.2, if $1/(m + 1) < \varepsilon < 1/m$. This correspondence implies the following:

(1) The interval $(0, 1)$ is generally divided into three zones: left, central, and right zones, with the central zone always present on the scheme, and the left one (the case $p = 0$) and/or the right one ($q = 0$) may be absent.
(2) The left zone consists of an even number $2p$ alternating Δ_1- and Δ_2-subintervals with lengths $(1 - m\varepsilon)$ and $((m + 1)\varepsilon - 1)$, respectively.
(3) The right zone consists of an even number $2q$ alternating Δ_1- and Δ_2-subintervals.
(4) Each of Δ_1-subintervals belonging to the left and right zones contains 2 points.
(5) Each of the Δ_2-subintervals of the left and right zones is empty, i.e. does not contain points.
(6) The central zone consists of an odd — $2(m - p - q) + 1$ — number of alternating Δ_1- and Δ_2-subintervals: $((m-p-q)+1)$ subintervals Δ_1 and $(m - p - q)$ subintervals Δ_2.
(7) Each of the Δ_1- and Δ_2-subintervals of the central zone contains one point.

The described scheme with partitioning of the interval $(0, 1)$ into subintervals of two types (hatched Δ_1 and open Δ_2) is not only a geometrical illustration of the possible random points allocation. With the help of such representation, the "continuous" initial problem reduces to "discrete", so we are able to solve it by combinatorial-probabilistic methods.

Note. *In the text, the following terminology is used. Each of the random points x_i thrown onto the interval $(0, 1)$ is named as "even" or "odd" depending on the parity (or oddness) of its index "i" in the ranked row $0 < x_1 < x_2 < \cdots < x_{2m+1} < 1$. In addition, based on belonging to the zones Δ_{LEFT}, Δ_{RIGHT}, or Δ_{CENTER},*

all interval points are divided into "left-zonal", "right-zonal", and "central zonal". This is introduced for convenience to facilitate the subsequent presentation when it is required (in particular, for the proof of Statement 7*) a more detailed analysis of the relative point's position. For greater clarity, all subintervals* Δ_1 *(see Figure* 10.2*) are hatched.*

Statement 1. If the parameters p and q are fixed, then the probability that after throwing of $n = 2m + 1$ points on the interval $(0, 1)$, exactly 2 $(p + q)$ points get into the joined zone $\Delta_{\text{LEFT}} \cup \Delta_{\text{RIGHT}}$, is

$$P_1 = C_{2m+1}^{2(p+q)}[(p+q)\varepsilon]^{2(p+q)}[1-(p+q)\varepsilon)]^{2(m-p-q)+1}.$$

Statement 2. The conditional probability that exactly 2 points get into each of the $(p+q)$ ε-segments of the joined zone $\Delta_{\text{LEFT}} \cup \Delta_{\text{RIGHT}}$ (if the joined zone contains $2(p + q)$ points) is

$$P_2 = \frac{(2(p+q))!}{(p+q)^{2(p+q)}2^{p+q}}.$$

Statement 3. The conditional probability that simultaneously in all ε-segments of the joined zone $\Delta_{\text{LEFT}} \cup \Delta_{\text{RIGHT}}$ both points got to each of them are located in the hatched sub-intervals Δ_1 is

$$P_3 = \left(\frac{1-m\varepsilon}{\varepsilon}\right)^{2(p+q)}.$$

Statement 4. The conditional probability that exactly $(m - p - q)$ points got into the unhatched part of the central zone Δ_{CENTER} (if this zone contains $2(m - p - q) + 1$ points) is

$$P_4 = C_{2(m-p-q)+1}^{m-p-q}\left[\frac{(m-p-q)((m+1)\varepsilon-1)}{1-(p+q)\varepsilon}\right]^{m-p-q} \times \left[\frac{((m-p-q)+1)(1-m\varepsilon)}{1-(p+q)\varepsilon}\right]^{m-p-q+1}.$$

Statement 5. The probability that $(m - p - q)$ central-zonal points got into the unhatched part of the central zone Δ_{CENTER} are

distributed strictly one in each of the $(m-p-q)$ unhatched Δ_2-segments is

$$P_5 = \frac{(m-p-q-1)!}{(m-p-q)^{m-p-q-1}}.$$

Statement 6. The probability that $((m-p-q)+1)$ points got into the hatched part of the central zone Δ_{CENTER} are distributed strictly one in each of the $((m-p-q)+1)$ hatched Δ_1-segments is

$$P_6 = \frac{(m-p-q)!}{(m-p-q+1)^{m-p-q}}.$$

Statement 7. The probability P_7 that simultaneously:

(a) Local coordinates of $(m+1)$ odd points, i.e. coordinates interpreted as the value of a random variable inside of the ε-segment, that contain each of odd points got (in other words, the local coordinates are the coordinates of the random point taken with module ε) form a monotonically increasing sequence (this condition just also provides that odd points are removed from each to other by a distance exceeding ε inside the interval $(0,1)$).
(b) Local coordinates p of even left-zoned points form a monotonically increasing sequence.
(c) Local coordinates q of even right-zonal points form a monotonically increasing sequence.
(d) Local coordinates $(m-p-q)$ of even central-zonal points form a monotonically increasing sequence

$$P_7 = \frac{(m+2)(m+3)-(p+q)(m+3)+pq}{(m+2)(m+3)} \times \frac{2^{p+q}}{(m+1)!p!q!(m-p-q)!}.$$

The final statement: Multiplying all the probabilities and performing the summation

$$\sum_{p=0}^{m}\sum_{q=0}^{m-p}(P_1 \times P_2 \times P_3 \times P_4 \times P_5 \times P_6 \times P_7),$$

after standard transformations and without serious difficulties, we obtain the formula

$$\begin{aligned} P_{2m+1,2}(\varepsilon) = {} & C_{2m+1}^{m+1}(1-m\varepsilon)^{m+1}(1-(m-1)\varepsilon)^{m} \\ & - 2C_{2m+1}^{m+2}(1-m\varepsilon)^{m+2}(1-(m-1)\varepsilon)^{m-1} \\ & + C_{2m+1}^{m+3}(1-m\varepsilon)^{m+3}(1-(m-1)\varepsilon)^{m-2}. \end{aligned} \tag{10.33}$$

It's correct in the range $1/(m+1) < \varepsilon < 1/m$.

So, the exact analytic solution of the formulated problem is found. Now, as it was planned earlier, it's time to give the necessary explanations and to show detailed proof (if it is required) for each of the statements formulated.

Explanation to Statement 1: Special explanations are not required: it's the usual Bernoulli scheme. $(2m+1)$ points are thrown on the interval $(0,1)$. The probability of "success" (getting into the zone $\Delta_{\text{LEFT}} \cup \Delta_{\text{RIGHT}}$) is $(p+q)\varepsilon$, the probability of "failure" is $1-(p+q)\varepsilon$.

Explanation to Statement 2: It's a simple combinatorial relation. It is required to find the probability $P_r(w)$ of the event that when $2w$ balls are thrown into w urns, there will be exactly 2 balls into each urn. Let $P2(N,r)$ be the probability of the event that if N balls are thrown on r urns, exactly 2 balls will get to the urn in advance chosen. It's obvious that

$$P2(2s,s) = C_{2s}^{2} \times \left(\frac{1}{s}\right)^{2} \times \left(1-\frac{1}{s}\right)^{2s-2}.$$

Since $\Pr(w) = P2(2w,w) \times P2(2w-2, w-1) \times P2(2w-4, w-2) \times \cdots \times P2(2,1)$, after dealing with the necessary substitutions and elementary transformations, we get that

$$\Pr(w) = \frac{(2w)!}{w^{2w}2^{w}}.$$

In conclusion, it is necessary to make substitution $w = p+q$.

Explanation to Statement 3: Each of the ε-segments forming the joined zone $\Delta_{\text{LEFT}} \cup \Delta_{\text{RIGHT}}$ consists of two subintervals: the subinterval Δ_1 of length $(1-m\varepsilon)$ and the subinterval Δ_2 of length $(\varepsilon - \Delta_1) = ((m+1)\varepsilon - 1)$. If each ε-segment contains two points,

then the probability that both of them got into the subinterval Δ_1 is equal to $\left(\frac{1-m\varepsilon}{\varepsilon}\right)^2$. Since the total number of ε-segments forming the zone $\Delta_{\text{LEFT}} \cup \Delta_{\text{RIGHT}}$ is $(p+q)$, then the probability that such a situation is the same in all ε-segments is $P_3 = \left(\frac{1-m\varepsilon}{\varepsilon}\right)^{2(p+q)}$.

Explanation to Statement 4: $(2m+1)$ points are thrown on the interval $(0,1)$. $2(p+q)$ of them got into the joined zone $\Delta_{\text{LEFT}} \cup \Delta_{\text{RIGHT}}$. So, $2(m-p-q)+1$ points got into the central zone Δ_{CENTER}. The length of the joined zone $\Delta_{\text{LEFT}} \cup \Delta_{\text{RIGHT}}$ is $(p+q)\varepsilon$, therefore, the length of the central zone Δ_{CENTER} is $1-(p+q)\varepsilon$. $(m-p-q)((m+1)\varepsilon-1)$ of central zone's length is reserved on $(m-p-q)$ unhatched subintervals Δ_2, and $((m-p-q)+1)(1-m\varepsilon)$ — on $((m-p-q)+1)$ hatched subintervals Δ_1. Therefore, the probability that exactly $(m-p-q)$ points will get into the unhatched part of the central zone Δ_{CENTER} (and, correspondingly, exactly $((m-p-q)+1)$ points get into its hatched part) is

$$P_4 = C^{m-p-q}_{2(m-p-q)+1} \left[\frac{(m-p-q)((m+1)\varepsilon-1)}{1-(p+q)\varepsilon}\right]^{m-p-q} \times \left[\frac{((m-p-q)+1)(1-m\varepsilon)}{1-(p+q)\varepsilon}\right]^{m-p-q+1}.$$

Explanation to Statement 5: It's not a complicated combinatorial relation, which can be proved in various ways, for example, by analogy with the above proof of Statement 2.

Explanation to Statement 6: This is a complete analogy with statements 2 and 5.

Explanation to Statement 7: The key moment in the proof of Statement 7 is a combinatorial "linguistic" problem, for its solution we needed to introduce the concept and find explicit expressions for generalized Catalan numbers. In brief words, the essence of the problem is here.

A problem with three-character words, leading to generalized Catalan numbers: From N_a characters "a", N_b characters "b" and N_c characters "c", different words of $(N_a+N_b+N_c)$ length are composed. It is necessary to determine the total number $W(N_a,N_b,N_c)$ of words that simultaneously satisfy two conditions:

(1) when viewing the word from left to right, the number of characters "b" encountered never exceeds the number of characters "a" encountered; (2) when viewing the word from right to left, the number of characters "c" encountered never exceeds the number of characters "a" encountered.

Additional information: In the problems presented in this chapter and related to the study of point images (in particular, finding the formulas $P_{2m+1,2}(\varepsilon)$), it is sufficient to get the solution of the formulated "three-characters" problem under the condition $N_a \geq N_b + N_c$. Such a solution can be obtained using a simple geometric model and double "mirror" reflection, it is shown in the following.

Solution. Let us consider different paths (see Figure 10.3) on a three-dimensional discrete lattice connecting the point $(0, 0, 0)$ with the point (N_a, N_b, N_c), in the coordinate system (X, Y, Z). We establish a one-to-one correspondence between these ways and words of $(N_a + N_b + N_c)$ length.

The movement from the current point (i, j, k) to the neighboring point $(i+1, j, k)$ corresponds to symbol "a", to the point $(i, j+1, k)$ to symbol "b", and to the point $(i, j, k+1)$ to symbol "c". Our task is to find the number of such paths from the point $(0, 0, 0)$ to the point (N_a, N_b, N_c) that do not intersect the plane $\mathscr{P}_1$ given by the equation $X - Y = 0$ (i.e. all the paths lie in a half-space $X \geq Y$) and also the plane $\mathscr{P}_2$ given by the equation $X - Z + N_{c-}N_a = 0$ (i.e. each of the paths lies not only in the half-space $X \geq Y$ but also in the half-space $N_{a-}X \geq N_{c-}Z$).

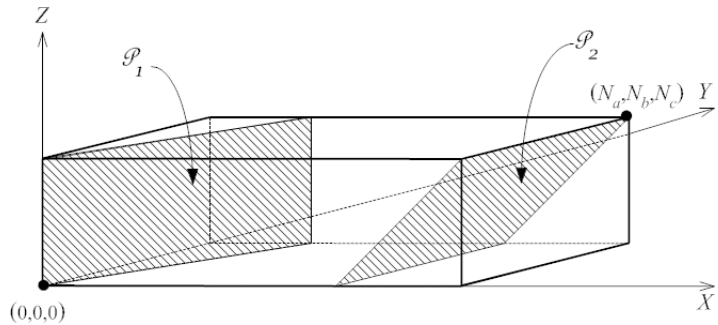


Figure 10.3. Parallelepiped with bounding planes $\mathscr{P}_1$ and $\mathscr{P}_2$.

It is possible to solve this reformulated problem in two ways. First, we can use the result obtained in [14]. Second, it is possible to apply twice the brilliant idea of mirror reflection, first proposed by the French mathematician D. Andre [1] to solve the ballot problem. Due to the greater clarity and absolute simplicity of getting the required result, we chose the second variant (note that under the condition $N_a \geq N_b + N_c$, the planes $\mathscr{P}_1$ and $\mathscr{P}_2$ do not intersect within the bounding parallelepiped presented in Figure 10.3 — this is important).

So, to find the solution to our task, we need to calculate the difference between the total number of paths connecting the point $(0, 0, 0)$ with the point (N_a, N_b, N_c), equal to

$$W_\Sigma = \frac{(N_a + N_b + N_c)!}{N_a! N_b! N_c!},$$

and the number of paths W_0 that intersect at least one of the planes $\mathscr{P}_1$ or $\mathscr{P}_2$. In turn, to calculate W_0, it is necessary to add the number of paths W_1 intersecting the plane $\mathscr{P}_1$ to the number of W_2 paths intersecting the plane $\mathscr{P}_2$ and subtract from the resulting sum the number of paths W_{12} intersecting both $\mathscr{P}_1$ and $\mathscr{P}_2$ planes (since they are counted twice in sum).

Let us first calculate the number of paths W_1. To do this, we use the following technique. We take any path intersecting the plane $\mathscr{P}_1$ and find the first-intersection point. Its coordinates obviously have the (u, u, v) form, and the next waypoint has $(u, u+1, v)$ coordinates. In the rest part of the path lying after the point $(u, u + 1, v)$, we do the mirror reflection with respect to the plane $Y = X + 1$, namely: each step along the X axis is replaced by a step along the Y axis, and vice versa, each step along the Y axis is replaced by a step along the X axis. It is easy to see that such a "corrected" path ends not at (N_a, N_b, N_c) point but at $(N_b{-}1, N_a + 1, N_c)$ point. Thus, every path that connects the point $(0, 0, 0)$ with the point (N_a, N_b, N_c) and intersects the plane $\mathscr{P}_1$, uniquely corresponds to a path that connects the point $(0, 0, 0)$ with the point $(N_b{-}1, N_a + 1, N_c)$. The converse statement is also true: each path from point $(N_b{-}1, N_a + 1, N_c)$ to point $(0, 0, 0)$ can be assigned to its own unique path that connects points $(0, 0, 0)$ and (N_a, N_b, N_c) and intersects the plane $\mathscr{P}_1$. To see this, first, you need to note that the points $(N_b{-}1, N_a + 1, N_c)$ and $(0, 0, 0)$ lie on different sides of the plane $Y = X + 1$ (according to

original task condition $N_b, N_c \leq N_a$) so that any path connecting the point $(N_b{-}1, N_a + 1, N_c)$ with the point $(0, 0, 0)$ inevitably intersects the plane $Y = X + 1$. Let us consider the fragment of the selected path from the point $(N_b{-}1, N_a + 1, N_c)$ to the first point of tangency with the plane $Y = X + 1$. Obviously, this fragment must end with such a link: $(u, u + 3, v) \to (u, u + 2, v) \to (u, u + 1, v)$, because otherwise the touch with the plane will not be the first. Now we replace the initial fragment of the path (from point $(N_b{-}1, N_a{+}1, N_c)$ to the first touch with the plane $Y = X + 1$ by its mirror reflection with respect to this plane), and the final fragment (from the point of the first point of tangency with the plane $Y = X + 1$ to the point $(0, 0, 0)$) remains unchanged. As a result, we get such a composite path that (1) begins at point $(N_b{-}1, N_a + 1, N_c)$, (2) ends at the point $(0, 0, 0)$, and (3) intersects the plane $\mathscr{P}_1$ given by the equation $Y = X$, since the link $(u, u + 3, v) \to (u, u + 2, v) \to (u, u + 1, v)$ under mirror reflection relative to the plane $Y = X + 1$ will go to the link $(u + 2, u + 1, v) \to (u + 1, u + 1, v) \to (u, u + 1, v)$.

So, we convert any path that connects the point $(0, 0, 0)$ with the point $(N_b{-}1, N_a + 1, N_c)$, into the path that connects $(0, 0, 0)$ and (N_a, N_b, N_c) points and at least once intersects the plane $\mathscr{P}_1$. So, we established the one-to-one correspondence between these two sets of paths. Therefore,

$$W_1 = \frac{((N_b - 1) + (N_a + 1) + N_c)!}{(N_b - 1)!(N_a + 1)!N_c!} = \frac{(N_a + N_b + N_c)!}{(N_a + 1)!(N_b - 1)!N_c!}.$$

Doing the same procedures and using the absolute symmetry of the problem for the parameters N_b and N_c, we find that

$$W_2 = \frac{(N_a + N_b + N_c)!}{(N_a + 1)!N_b!(N_c - 1)!}.$$

Now it becomes obvious that for calculation the number of paths W_{12} that intersect both planes $\mathscr{P}_1$ and $\mathscr{P}_2$, we need to use the mirroring procedure twice. This is done as follows. The path that intersects both planes $\mathscr{P}_1$ and $\mathscr{P}_2$ can be schematically represented in this form:

$$\begin{aligned}(0, 0, 0) &\to \cdots \to (x_1, x_1, z_1) \to (x_1, x_1 + 1, z_1) \to \cdots \\ &\to (x_2, y_2, x_2 + N_{c-}N_a) \to (x_2 + 1, y_2, x_2 + N_{c-}N_a) \\ &\to \cdots \to (N_a, N_b, N_c).\end{aligned}$$

The first mirroring we do over the initial part of the path that begins at the point $(0,0,0)$ and ends at the point (x_1, x_1+1, z_1), which is the first point of the path lying in the half-space $X < Y$ (i.e. on the other side of the plane $\mathscr{P}_2$). Moving back from the point (x_1, x_1+1, z_1), each time we take a backward step along the Y axis instead of a backward step along the X axis, and vice versa — instead of a backward step along the Y axis, we take a backward step along the X axis. It's easy to see after such a transformation the "corrected" path will not start at the $(0,0,0)$ point but at the $(-1,+1,0)$ point.

The second mirroring we do for the final segment of the path starts at $(x_2+1, y_2, x_2+N_{c-}N_a)$ and ends at (N_a, N_b, N_c). For clarity, as before, we assume that the point $(x_2+1, y_2, x_2+N_{c-}N_a)$ is the first point of the path lying on the other side of the plane $\mathscr{P}_2$ (in general, there may be several such points in one path). The transformation of the final segment of the path is done in the following way: the step along the X axis is replaced by the step along the Z axis, and the steps along the Z axis are replaced by the steps along the X axis. As a result, the end point of the path is not the point (N_a, N_b, N_c) but the point $(N_a+1, N_b, N_{c-}1)$.

So, any path connects the point $(0,0,0)$ with to the point (N_a, N_b, N_c) and intersecting both planes $\mathscr{P}_1$ and $\mathscr{P}_2$, uniquely corresponds to a path connecting the point $(-1,+1,0)$ with to the point $(N_a+1, N_b, N_{c-}1)$. The converse statement that any path connects the point $(-1,+1,0)$ with the point $(N_a+1, N_b, N_{c-}1)$ can be put in one-to-one correspondence to the path connects the point $(0,0,0)$ with the point (N_a, N_b, N_c) and intersecting both planes $\mathscr{P}_1$ and $\mathscr{P}_2$ can be easy proved if, first, we take into consideration that the points $(-1,+1,0)$ and $(N_a+1, N_b, N_{c-}1)$ lie in different half-spaces relative to both planes $\mathscr{P}_1$ and $\mathscr{P}_2$. This means that the path connecting them will inevitably intersect both these planes. Next, we need to do two mirror transformations (with the initial and final segments of the path, respectively) that are inverse to the transformations described above and to make sure that the transformed path connects the point $(0,0,0)$ with the point (N_a, N_b, N_c) and intersects both planes $\mathscr{P}_1$ and $\mathscr{P}_2$. As a result, we establish a one-to-one correspondence between the set of paths connecting the point $(0,0,0)$ with the point (N_a, N_b, N_c) and intersecting both planes $\mathscr{P}_1$ and $\mathscr{P}_2$,

and the set of paths leading from the point $(-1, +1, 0)$ to the point $(N_a + 1, N_b, N_c{-}1)$. Therefore,

$$W_{12} = \frac{(N_a + N_b + N_c)!}{(N_a + 2)!(N_b - 1)!(N_c - 1)!}.$$

As a result, the solution of the formulated problem on the number of three-character words (under the condition $N_a \geq N_b + N_c$) is written as follows:

$$\begin{aligned} &W(N_a, N_b, N_c) \\ &\quad = W_\Sigma - W_1 - W_2 + W_{12} \\ &\quad = \frac{(N_a + N_b + N_c)!}{N_a! N_b! N_c!} - \frac{(N_a + N_b + N_c)!}{(N_a + 1)!(N_b - 1)! N_c!} \\ &\qquad - \frac{(N_a + N_b + N_c)!}{(N_a + 1)! N_b!(N_c - 1)!} + \frac{(N_a + N_b + N_c)!}{(N_a + 2)!(N_b - 1)!(N_c - 1)!} \\ &\quad = \frac{(N_a + N_b + N_c)!}{N_a! N_b! N_c!} \left[1 - \frac{N_b + N_c}{N_a + 1} + \frac{N_b N_c}{(N_a + 1)(N_a + 2)}\right]. \end{aligned} \tag{10.34}$$

Generally speaking, using the approach described in [14], one can find the general (without the restriction $N_a \geq N_b + N_c$) solution of the three-character word problem. It can be found as the number of random paths in Weyl's chambers:

$$\begin{aligned} &W(N_a, N_b, N_c) \\ &\quad = \frac{(N_a + N_b + N_c)!}{N_a! N_b! N_c!} - \frac{(N_a + N_b + N_c)!}{(N_a + 1)!(N_b - 1)! N_c!} \\ &\qquad - \frac{(N_a + N_b + N_c)!}{(N_a + 1)! N_b!(N_c - 1)!} + \frac{(N_a + N_b + N_c)!}{(N_a + 2)!(N_b - 1)!(N_c - 1)!} \\ &\qquad + \frac{(N_a + N_b + N_c)!}{(N_b + N_c - N_a - 2)!(N_a + 1)!(N_a + 1)!} \\ &\qquad - \frac{(N_a + N_b + N_c)!}{(N_b + N_c - N_a - 2)! N_a!(N_a + 2)!} \end{aligned}$$

We called the numbers $W(N_a, N_b, N_c)$ "the generalized three-dimensional Catalan numbers" meaning that the received formula

really generalizes classical Catalan sequence known from many applications and obtained from $W(N_a, N_b, N_c)$ when $N_c = 0$ and $N_b = N_a$. So, the problem with three-character words is solved, and the explicit form of the generalized Catalan numbers is found.

Returning to the mathematical basis of the probability P_7 given in Statement 7, we note that in our case symbols "a" correspond to $(m+1)$ odd points (i.e., $N_a = m+1$), the symbols "b" correspond to p even left-zonal points (i.e., $N_b = p$), and the symbols "c" correspond to q even right-zonal points (i.e., $N_c = q$). Thus, the number of "successful" placement variants, when the local coordinates of odd, even left-zonal, and even right-zonal points satisfy the constraints of the original problem, is

$$\frac{(m+p+q+1)!}{(m+1)!p!q!} \times \left[1 - \frac{p+q}{m+2} + \frac{pq}{(m+2)(m+3)}\right]$$

This value must be normalized over the total number of different placement variants of $(m+p+q+1)$ random variables

$$\frac{(m+p+q+1)!}{2^{p+q}}.$$

The coefficient 2^{p+q} in the denominator of the last expression represents the fact that in each of the Δ_1-segments of the left zone and in each of the Δ_1-segments of the right zone (total number of all such segments is $(p+q)$), compulsory ordering occurs: from two points got into every Δ_1-segments, even left-handed point has always larger coordinates and even right-zonal point has smaller coordinates.

The cofactor $1/(m-p-q)!$ included in the final formula for the probability P_7 characterizes the fact that the local coordinates of $(m-p-q)$ even central-zonal points that got in each of the Δ_2-segments of the central zone must form a monotonically increasing sequence not connected with the coordinates of all other points (moreover, such a connection is impossible, since the local coordinates of even central-zonal points are random variable uniformly distributed in the interval Δ_2 with length $((m+1)\varepsilon - 1)$, while the local coordinates of all other points — odd, even left-zonal, and even right-zonal — are values of a random variable uniformly distributed in the interval Δ_1, with a length $(1 - m\varepsilon)$.

Explanation of the final statement: After multiplying the probabilities and performing the simplifications, we get

$$\begin{aligned} P_{2m+1,2}(\varepsilon) &= \sum_{p=0}^{m}\sum_{q=0}^{m-p}(P_1\times P_2\times P_3\times P_4\times P_5\times P_6\times P_7) \\ &= \sum_{p=0}^{m}\sum_{q=0}^{m-p}\left\{\frac{(2m+1)!}{(m+1)!(m-p-q)!p!q!}\right. \\ &\quad\times\frac{(m+2)(m+3)-(p+q)(m+3)+pq}{(m+2)(m+3)} \\ &\quad\left.\times(1-m\varepsilon)^{m+p+q+1}((m+1)\varepsilon-1)^{m-p-q}\right\}. \end{aligned} \tag{10.35}$$

Further transformations of (10.35) are

$$\begin{aligned} &\sum_{p=0}^{m}\sum_{q=0}^{m-p}\left\{\frac{(2m+1)!}{(m+1)!(m-p-q)!p!q!}\right. \\ &\quad\times\frac{(m+2)(m+3)-(p+q)(m+3)+pq}{(m+2)(m+3)} \\ &\quad\left.\times(1-m\varepsilon)^{m+p+q+1}((m+1)\varepsilon-1)^{m-p-q}\right\} \\ &= \frac{(2m+1)!}{(m+3)!}\sum_{p=0}^{m}\left[\frac{1}{p!}\sum_{q=0}^{m-p}\left\{\frac{(m+2)(m+3)-(p+q)(m+3)+pq}{(m-p-q)!q!}\right.\right. \\ &\quad\left.\left.\times(1-m\varepsilon)^{m+p+q+1}((m+1)\varepsilon-1)^{m-p-q}\right\}\right] \\ &= \frac{(2m+1)!}{(m+1)!}\sum_{p=0}^{m}\left[\frac{(1-m\varepsilon)^{m+p+1}}{p!(m-p)!}\sum_{q=0}^{m-p}\left\{\frac{(m-p)!}{(m-p-q)!q!}\right.\right. \\ &\quad\left.\left.\times(1-m\varepsilon)^{q}((m+1)\varepsilon-1)^{m-p-q}\right\}\right] \end{aligned}$$

$$
\begin{aligned}
&- \frac{(2m+1)!}{(m+2)!} \sum_{p=1}^{m} \left[\frac{(1-m\varepsilon)^{m+p+1}}{(p-1)!(m-p)!} \sum_{q=0}^{m-p} \left\{ \frac{(m-p)!}{(m-p-q)!q!} \right.\right. \\
&\left.\left. \times (1-m\varepsilon)^{q}((m+1)\varepsilon-1)^{m-p-q} \right\} \right] \\
&- \frac{(2m+1)!}{(m+2)!} \sum_{p=0}^{m} \left[\frac{(1-m\varepsilon)^{m+p+2}}{p!(m-p-1)!} \sum_{q=1}^{m-p} \left\{ \frac{(m-p-1)!}{(m-p-q)!(q-1)!} \right.\right. \\
&\left.\left. \times (1-m\varepsilon)^{q-1}((m+1)\varepsilon-1)^{m-p-q} \right\} \right] \\
&+ \frac{(2m+1)!}{(m+3)!} \sum_{p=1}^{m} \left[\frac{(1-m\varepsilon)^{m+p+2}}{(p-1)!(m-p-1)!} \sum_{q=1}^{m-p} \left\{ \frac{(m-p-1)!}{(m-p-q)!(q-1)!} \right.\right. \\
&\left.\left. \times (1-m\varepsilon)^{q-1}((m+1)\varepsilon-1)^{m-p-q} \right\} \right] \\
&= \frac{(2m+1)!}{m!(m+1)!} \times (1-m\varepsilon)^{m+1}(1-(m-1)\varepsilon)^{m} \\
&- \frac{(2m+1)!}{(m-1)!(m+2)!} \times (1-m\varepsilon)^{m+2}(1-(m-1)\varepsilon)^{m-1} \\
&- \frac{(2m+1)!}{(m-1)!(m+2)!} \times (1-m\varepsilon)^{m+2}(1-(m-1)\varepsilon)^{m-1} \\
&+ \frac{(2m+1)!}{(m-2)!(m+3)!} \times (1-m\varepsilon)^{m+3}(1-(m-1)\varepsilon)^{m-2} \\
&= C_{2m+1}^{m+1}(1-m\varepsilon)^{m+1}(1-(m-1)\varepsilon)^{m} \\
&- 2C_{2m+1}^{m+2}(1-m\varepsilon)^{m+2}(1-(m-1)\varepsilon)^{m-1} \\
&+ C_{2m+1}^{m+3}(1-m\varepsilon)^{m+3}(1-(m-1)\varepsilon)^{m-2}.
\end{aligned}
$$

Thus, the correctness of the formula (10.33) is proved. So, the solution to the problem formulated at the beginning of Section 10.3.2 is complete.

10.3.3 *Combinatorial-linguistic problems concerned with the random sampled structures which lead to classical and generalized Catalan numbers*

Problem 1 (leading to the classical Catalan numbers). In the formation of a word, N characters "a" and N characters "b" are participated. How many of these words exist if when viewing the word from left to right, the number of encountered characters "b" never exceeds the number of encountered characters "a"?

Decision:

$$\mathrm{WORD1}(N) = \frac{(2N)!}{N!N!} - \frac{(2N)!}{(N+1)!(N-1)!} = \frac{1}{N+1}\binom{2N}{N}. \tag{10.36}$$

Formula (10.36) is known from Leonhard Euler's work (many applied probabilistic problems lead to it).

Problem 2. In the formation of the word, N_a characters "a" and N_b characters "b" are participated. How many of these words exist if when viewing the word from left to right, the number of encountered characters "b" never exceeds the number of "a" characters encountered? (It is assumed that $N_a \geq N_b$.)

Decision:

$$\mathrm{WORD2}(N_a, N_b) = \frac{(N_a+N_b)!}{N_a!N_b!} - \frac{(N_a+N_b)!}{(N_a+1)!(N_b-1)!}. \tag{10.37}$$

This is the simplest extension of the Catalan numbers. For $N_a = N_b$, we have the classical Catalan sequence. With the help of this extension, for example, it is very simple to obtain a solution to the classical Bertrand's ballot problem: "In an election where candidate A receives N_a votes and candidate B receives N_b votes with $N_a > N_b$, what is the probability that A will be strictly ahead of B throughout the count?"

It is obvious that the first ballot paper must be for candidate A, and for the remaining votes, it is sufficient to fulfill a weak inequality

in candidate A advantage. So, to calculate the number of election outcomes, when candidate A is always ahead of B, it is only necessary to replace N_a by $N_a - 1$ in formula (10.37):

$$\begin{aligned}\mathrm{WORD2}(N_a - 1, N_b) &= \frac{(N_a - 1 + N_b)!}{(N_a - 1)!N_b!} - \frac{(N_a - 1 + N_b)!}{N_a!(N_b - 1)!} \\ &= \frac{N_a - N_b}{N_a + N_b}\binom{N_a + N_b}{N_a}.\end{aligned}$$

As the total number of the election outcomes when the candidates receive N_a and N_b votes is $\binom{N_a + N_b}{N_a}$, the probability for candidate A to be ahead of B throughout the count is $\frac{N_a - N_b}{N_a + N_b}$.

Problem 3. In the formation of the word, N_a characters "a" and N_b characters "b" are participated. How many of these words exist the difference between the number of encountered characters "b" and the number of encountered characters "a" (when viewing the word from left to right) doesn't exceed k? (It is assumed $k \geq 0$ and $N_a + k \geq N_b$.)

Decision:

$$\mathrm{WORD3}_k(N_a, N_b) = \frac{(N_a + N_b)!}{N_a!N_b!} - \frac{(N_a + N_b)!}{(N_a + k + 1)!(N_b - k - 1)!}.$$

This is a further extension of the Catalan numbers. Note that for $k = 0$, problem №3 is equivalent to problem №2, and for $N_a = N_b$ and $k = 0$, we get problem №1.

Problem 4. With the help of the 6-character alphabet {"a", "b", "c", "d', "e", "f"} , various words of length $2m$ are composed in such a manner that

$$\begin{aligned}&N_a + N_d + N_e = m; \quad N_b + N_c + N_f = m; \\ &N_a + N_c \leq m,\end{aligned}$$

where $N_a, N_b, N_c, N_d, N_e,$ and N_f are the number of characters "a", "b", "c", "d", "e", and "f" used in this word, respectively. At these restrictions, it is required to find the total number

$WORD4_m(N_a, N_b, N_c, N_d, N_e, N_f)$ of words that simultaneously satisfy three conditions:

(1) When viewing from left to right, the number of encountered characters "b" never exceeds the number of encountered characters "a".
(2) When viewing from right to left, the number of encountered characters "d" never exceeds the number of encountered characters "c".
(3) When viewing from left to right, the difference between the number of encountered characters "f" and the number of encountered characters "e" never exceeds $(N_a - N_b)$.

Decision:

Using the classical method of mirror reflection of Andre [1], we found the solution of the formulated problem:

$$\begin{aligned}
&\mathrm{WORD4}_m(N_a, N_b, N_c, N_d, N_e, N_f)\\
&\quad = (2m)! \times \left\{ \frac{1}{(m-(N_b+N_c))!(m-(N_a+N_d))!} \right.\\
&\qquad \left. - \frac{1}{(m+1-(N_b+N_d))!(m-1-(N_a+N_c))!} \right\}\\
&\qquad \times \left\{ \frac{1}{\mathrm{N}_a!N_b!} - \frac{1}{(N_a+1)!(N_b-1)!} \right\}\\
&\qquad \times \left\{ \frac{1}{\mathrm{N}_c!N_d!} - \frac{1}{(N_c+1)!(N_d-1)!} \right\}.
\end{aligned}$$

Problem 5. With the help of the 6-character alphabet {"a", "b", "c", "d", "e", "f"}, various words of length $2m$ are composed in such a manner that

$$\begin{aligned}
&N_a + N_d + N_e = m; \quad N_b + N_c + N_f = m;\\
&N_a + N_c \geq m + 1,
\end{aligned}$$

where N_a, N_b, N_c, N_d, N_e, and N_f are the number of characters "a", "b", "c", "d", "e", and "f" used in this word, respectively. At these restrictions, it is required to find the total number

$WORD5_m(N_a, N_b, N_c, N_d, N_e, N_f)$ of words that simultaneously satisfy three conditions:

(1) When viewing from left to right, the number of characters "b" encountered never exceeds the number of characters "a" encountered.
(2) When viewing from right to left, the number of characters "d" encountered never exceeds the number of characters "c" encountered.
(3) When viewing from left to right, the difference between the number of encountered characters "a" and the number of encountered characters "c" never exceeds $(m + 1 - N_c)$.

Decision:

Using the approach from [14], we found the solution of this problem:

$$\begin{aligned}
&\mathrm{WORD5}_m(N_a, N_b, N_c, N_d, N_e, N_f) \\
&\quad = \frac{(2m)!}{N_e!N_f!} \times \left\{ \left[\frac{1}{N_a!N_b!N_c!N_d!} - \frac{1}{N_a!N_b!(N_c+1)!(N_d-1)!} \right] \right. \\
&\qquad + \left[\frac{1}{(N_a+1)!(N_b-1)!(N_c+1)!(N_d-1)!} \right. \\
&\qquad \left. - \frac{1}{(N_a+1)!(N_b-1)!N_c!N_d!} \right] \\
&\qquad + \left[\begin{array}{c} \dfrac{1}{(m+3)!N_b!(N_d-1)!(N_a+N_c-(m+2))!} \\ -\dfrac{1}{(m+2)!N_b!N_d!(N_a+N_c-(m+2))!} \end{array} \right] \\
&\qquad + \left. \left[\begin{array}{c} \dfrac{1}{(m+3)!(N_b-1)!N_d!(N_a+N_c-(m+2))!} \\ -\dfrac{1}{(m+4)!(N_b-1)!(N_d-1)!(N_a+N_c-(m+2))!} \end{array} \right] \right\}.
\end{aligned}$$

Using the explicit form of the generalized Catalan numbers given in Section 10.3, we succeeded in establishing and proving a series of new previously unknown analytic relations describing probability $P_{n,2}(\varepsilon)$. In particular, we have proved [28] that for even values $n = 2m$, in the range $1/(m+1) < \varepsilon < 1/m$, the formula is correct

$$\begin{aligned} P_{2m,2}(\varepsilon) = {} & C_{2m}^{m}(1-(m-1)\varepsilon)^{2m} - C_{2m}^{m-1}(1-(m-1)\varepsilon)^{2m} \\ & - C_{2m}^{m-2}(1-m\varepsilon)^{m+2}(1-(m-2)\varepsilon)^{m-2} \\ & + 2C_{2m}^{m-3}(1-m\varepsilon)^{m+3}(1-(m-2)\varepsilon)^{m-3} \\ & - C_{2m}^{m-4}(1-m\varepsilon)^{m+4}(1-(m-2)\varepsilon)^{m-4} \end{aligned} .$$

We are sure that the accumulation of particular results carried out by means of programs of machine analytics and the generalized Catalan numbers will lead to the establishment of the general analytical dependence $P_{n,k}(\varepsilon)$.

10.4 Optimal-Speed Algorithms for Localization of Random Point-Pulse Sources

In this section, we present methods and algorithms relating to the analysis of dynamically changing discrete random fields. In Section 10.4.1, we study algorithms of time-optimal search for Poisson sources indicating themselves by generation instant delta pulses at random time points. Time-optimal search algorithm should, generally, meet one of two requirements: either to minimize the total search effort necessary to detect the object or to maximize the full detection probability in the presence of a limited search effort.

We understand the point-pulse source as an object with negligible small angular dimensions (a mathematical point) that is placed randomly on the x-axis with *a priori* probability density function $f(x)$ and emitting infinitely short pulses (δ-functions) with a Poisson intensity λ. Thus, the time interval between pulses is random variable t with an exponential density function $h(t) = \lambda\exp(-\lambda t)$. The search for the object is performed by a recording device, whose "viewing window" is able to be reconstructed in any way in time. The pulse is fixed if the active object produced the pulse is in the "viewing window" of the recording device. Otherwise, the pulse is

considered to be missed. As the pulse is registered, the window narrows, and the source position becomes more precise. It is required to search for the source during the minimum (statistically) time with ε accuracy.

In the final part of Section 10.4.2, the adaptation of time-optimal search methods to systems with several recording devices is described. The expansion of the proposed optimal search methods in the case of two-dimensional and three-dimensional domains is an independent task and is not considered in this chapter.

10.4.1 *Programs and algorithms for optimal-speed search of single point-pulse source*

(a) One-step search algorithms: Introducing the binary function $u(x,t)$ describing the viewing window at time t, we obtain the relation for the average time from the beginning of the search to recording the first pulse:

$$\langle\tau\rangle = \lambda \int_0^\infty dt \int_0^\infty dx \left[t f(x) u(x,t) \exp\left(-\lambda \int_0^t u(x,\xi) d\xi \right) \right]. \tag{10.38}$$

For the random *a priori* distribution of the source on the axis x, the construction of even one-step (ending after registering the first pulse) optimal-time search procedure causes considerable difficulties. In one-step *periodic* search algorithms, the relative load $\varphi(x)$ per point x (i.e., the relative time the point is present in the viewing window) remains constant throughout the search time. With this approach, the task is to find the function that minimizes the average search time

$$\langle\tau\rangle = \frac{1}{\lambda} \int \frac{f(x)}{\phi(x)} dx \tag{10.39}$$

provided that

$$\int \phi(x) dx = \varepsilon, \tag{10.39a}$$

$$0 \le \phi(x) \le 1. \tag{10.39b}$$

Using the method of undetermined Lagrange multipliers, we search for the function $\varphi(x)$ that minimizes the expression

$$\int \left[\frac{f(x)}{\phi(x)} + \mu\phi(x)\right] dx.$$

Further, differentiating with respect to φ and taking into account the constraint (10.39a), we get

$$\phi(x) = \frac{\varepsilon\sqrt{f(x)}}{\int \sqrt{f(x)}dx} \tag{10.40}$$

If at the same time the condition (10.39b) is not violated, then the function (10.40) is a solution to the formulated extreme problem. If there are such ranges of x where the solution $\varphi(x) > 1$, then inside of these ranges it is necessary to set $\varphi(x) = 1$, and for the remaining points, it is required to recalculate the undefined multiplier μ under already changed conditions (10.39) and (10.39a). After this, as an optimal search strategy, any binary function $u(x,t)$ can be chosen that satisfies the relations

$$\int u(x,t)dx = \varepsilon; \quad \int u(x,\xi)d\xi = \phi(x)t.$$

In the general case, the construction of the optimal (not necessarily periodic) one-step search algorithm for an unknown Poisson source is connected with finding such a function $\varphi(x,t)$ — the relative load on the point x at a time t that minimizes the average search time

$$\langle\tau\rangle = \int dt \int dx f(x) \exp\left(-\lambda \int_0^t \phi(x,\xi)d\xi\right) \tag{10.41}$$

provided that

$$\int \phi(x,t)dx = \varepsilon \quad \text{for any } t, \tag{10.41a}$$

$$0 \le \phi(x,t) \le 1. \tag{10.41b}$$

To simplify further calculations, we introduce a function $\alpha(x,t) = \int_0^t \phi(x,\xi)d\xi$ corresponding to the total time spent by the point x in the viewing window from the beginning of the search to time t. To

take into account the constraints (10.41a) and (10.41b), we introduce the undetermined Lagrange multiplier $\mu(t)$. Then the problem of finding the optimal strategy reduces to finding the function $\alpha(x,t)$ that minimizes the functional

$$\int dt \int dx \left[\exp(-\lambda\alpha(x,t)f(x) + \mu(t)\alpha(x,t)\right] \tag{10.42}$$

provided that

$$\int_{-\infty}^{\infty} \alpha(x,t)dx = \varepsilon t, \tag{10.42a}$$

$$0 \le \alpha(x,t) \le t. \tag{10.42b}$$

The solution of this variational problem is the function

$$\alpha(x,t) = \begin{cases} 0, \dfrac{1}{\lambda}\ln\dfrac{\lambda f(x)}{\mu(t)} < 0; \\ \dfrac{1}{\lambda}\ln\dfrac{\lambda f(x)}{\mu(t)}, 0 \le \dfrac{1}{\lambda}\ln\dfrac{\lambda f(x)}{\mu(t)} \le t; \\ t, t < \dfrac{1}{\lambda}\ln\dfrac{\lambda f(x)}{\mu(t)}, \end{cases} \tag{10.43}$$

where $\mu(t)$ is determined from the relation (10.42a). The optimal search strategy $u(x,t)$ must belong to the class of binary functions. It is set by the equations

$$\int_0^t u(x,\xi)d\xi = \alpha(x,t); \quad \int u(x,t)dx = \varepsilon.$$

The use of optimal search algorithms in practice faces certain difficulties. The fact is both proposed algorithms of optimal one-step search, when *a priori* density function of the signal source differs from the constant, cannot be physically realized (as a rule) by moving a simply connected viewing window with a width ε. Therefore, in real search procedures, one-step procedure is advisable to do in according to the following scheme.

Preliminarily, the interval $(0, L)$ is divided into ε-step discrete and *a priori* given density $f(x)$ is "stepwise" approximated on each of them. In this case, ε is considered to be sufficiently small so that the

variation of the function $f(x)$ within one discrete can be neglected. Search (in accordance with the optimal one-step detection procedure) should begin only with the highest "peak" density inspection, then after a time t_1 the window is alternately set to the areas under the two highest "peaks", then after time t_2, further inspection is done for three sections, and so on. All the moments t_i are exactly determined by the described finding method for optimal function $\alpha(x,t)$.

It should be noted that the discussed search plan assumes that the intensity of the source λ is known in advance. If such *a priori* information is not available, a periodic procedure can be recommended that does not depend on this intensity. In accordance with it, the integrals of the density $f(x)$ in each of the discrete must be calculated. If there are m discretes, and its squares-integrals are $P_1, P_2, \ldots, P_m$, then the viewing window should cyclically "run through" all the discretes with relative load $\beta_i = \sqrt{P_i}/\sum_{j=1}^{m}\sqrt{P_j}$ $(i = 1, \ldots, m)$, respectively. These values β_i are easily obtained if we again apply the method of undetermined Lagrange multipliers to minimize the average search time

$$\langle\tau\rangle = (1/\lambda)(P_1/\beta_1 + P_2/\beta_2 + \cdots + P_m/\beta_m) \tag{10.44}$$

provided that

$$\beta_1 + \cdots + \beta_m = 1. \tag{10.45}$$

(b) Multistep search algorithms: If we are not limited to one-step procedures, but consider the search algorithm as a multistep process (that ends after nth pulse registration), then the optimal strategy should deliver a minimum to the functional

$$\begin{aligned}\langle\tau\rangle = \sum_{k=1}^{n}\lambda^k \int_0^\infty dx f(x) \int\cdots\int t_k \\ \times\left\{\prod_{l=1}^{k}\left[dt_l u_l\left(x, \sum_{m=1}^{l} t_m | t_1, \ldots, t_{l-1}\right)\right.\right. \\ \left.\left.\times\exp\left(-\lambda\int_{\sum_{m=1}^{l-1} t_m}^{\sum_{m=1}^{l} t_m} u_l(x,\xi|t_1,\ldots t_{l-1})d\xi\right)\right]\right\},\end{aligned} \tag{10.46}$$

provided that

$$\int u_n(x,t|t_1,\ldots,t_{n-1})dx = \varepsilon. \tag{10.47}$$

Here $u_i(x,t|t_1,\ldots t_{i-1})$ is the search strategy at the ith step provided that the intervals between the first $(i-1)$ pulses were $t_1,t_2,\ldots,t_{i-1}$, respectively. In the general case, finding the optimal strategy $u(x,t)$ that minimizes the functional (10.46) is not possible. At the same time, for an important special case, $f(x) = \text{const}$, the analytic solution is rather simple. Let

$$f(x) = \begin{cases} \dfrac{1}{L}, & x \in (0,L) \\ 0, & x \notin (0,L), \end{cases}$$

i.e. there is no *a priori* information about the placement of the source within the interval $(0,L)$. Obviously, in the first step, the search load should be equally distributed between all the points $x \in (0,L)$. It is possible to make such a load, for example, by television scanning of the whole interval $(0,L)$ by the aperture with width l_1 (to avoid edge effects, we consider the end of the interval closed to its beginning, so, a circle with length L is scanned instead of the interval). When registering a pulse, the search continues inside the window with width l_1 using another aperture with width l_2 ($l_2 < l_1$). If we discuss an n-step search, then at the last step scanning it is carried out by the aperture with width ε (this is dictated by the conditions of the task). Then the average search time is

$$\langle\tau\rangle = (1/\lambda)(L/l_1 + l_1/l_2 + \cdots + l_{n-1}/\varepsilon). \tag{10.48}$$

For a fixed n, it's possible to find optimal values that minimize the expression (10.48):

$$L/l_1 = l_1/l_2 = \cdots = l_{n-1}/\varepsilon = (L/\varepsilon)^{1/n}. \tag{10.49}$$

Then the average time of the optimal n-step search is

$$\langle\tau_n\rangle_{\text{opt}} = (n/\lambda)(L/\varepsilon)^{1/n}. \tag{10.50}$$

Now (from the expression (10.50)) we can find the optimal number of steps n minimizes the average search time. Since the function

$xa^{1/x}$ for $a > 1$ has only one minimum point ($x = \ln a$), the optimal value n_{opt} is always either entier($\ln(L/\varepsilon)$) or entier($\ln(L/\varepsilon)$) + 1. When $L/\varepsilon \to \infty$, it is possible to consider $n_{\text{opt}} \cong \ln(L/\varepsilon)$. Therefore, we have asymptotic relations:

$$L/l_1 = l_1/l_2 = \cdots = l_{n-1}/\varepsilon = e. \tag{10.51}$$

$$\langle \tau_n \rangle_{\text{opt}} = (n_{\text{opt}}/\lambda)(L/\varepsilon)^{1/n_{\text{opt}}} = (e/\lambda)\ln(L/\varepsilon). \tag{10.52}$$

So, a multistep procedure brings a gain in comparison with a one-step search (for a one-step search in the case under consideration, the average detection time is $\langle \tau \rangle_{\text{opt}} = L/(\lambda\varepsilon)$, and this gain increases unlimitedly with the ratio L/ε increasing. Now we can compare the constructed optimal search procedure with some simplified algorithms. For example, if the search is organized according to the principle of a dichotomy, then the average time of the object search is

$$\langle \tau \rangle_2 = (2/\lambda)\log_2(L/\varepsilon) = (2/\lambda \ln 2)\ln(L/\varepsilon). \tag{10.53}$$

Dichotomous search has (in comparison with the optimal procedure) a small ($\approx 6\%$) loss in time. Trichotomic search is even closer to the optimal procedure: initial interval $(0, L)$ is divided into three subintervals, then the subinterval where the pulse is fixed, in turn, is also divided into three subintervals, etc. This procedure loses to the optimal procedure only

$$(3/\ln 3 - e)/e \approx 0,4\%.$$

It is natural to expect that in the case of arbitrariest *a priori* distribution $f(x)$ the multistep search procedure (in comparison with one-step search) can bring a significant gain in time, especially for large values of L/ε. Since minimization of the functional (10.46) under the constraint (10.47) in each concrete case is a very difficult problem, multistep *periodic* search procedure seems to be more real from the practical realization point of view. In the first step, the interval $(0, L)$ is divided into three parts (three parts are chosen because for a uniform distribution $f(x)$ this procedure is closest to the optimal one). Then the values P_1, P_2, P_3 are calculated

$$P_1 = \int_0^{L/3} f(x)dx; \quad P_2 = \int_{L/3}^{2L/3} f(x)dx; \quad P_3 = \int_{2L/3}^{L} f(x)dx.$$

For any time interval Δt, the viewing window with width $L/3$ will periodically "run through" all three sections in such a way that $\Delta t_1 + \Delta t_2 + \Delta t_3 = \Delta t$, where $\Delta t_i/\Delta t = \beta_i$ is the relative viewing window presence time on each of the subintervals $(0, L/3)$, $(L/3, 2L/3)$, $(2L/3, L)$. When the pulse is registered, the search procedure continues similarly on the subinterval (segment) where the pulse is fixed (i.e., this segment is divided again into three parts, the coefficients $\beta_1, \beta_2, \beta_3$ are recalculated, etc.). In the first step, their values are equal $\sqrt{P_i}/\sum_{j=1}^{3}\sqrt{P_j}$ $(i = 1, 2, 3)$. The universality of the proposed procedure is also in the fact that its realization does not require *a priori* information on the source intensity.

10.4.2 *Systems with multiple receivers: Optimal-speed algorithms for random point-pulse sources search*

Problems and algorithms for the optimal search of random sources that are discussed in this paragraph arise in many scientific and technical fields, in particular, in classical disciplines such as reliability theory and mathematical communication theory. Similar studies are necessary for the development of methods for tech troubleshooting, appearing in a form of alternating equipment failure; in astrophysics, these problems are encountered in the search of pulsating radiation sources; in modern sections of computational mathematics these methods are required to develop algorithms for detecting low-contrast and small-sized objects on aerospace images, and, for example, in signal theory, the same methods are used to estimate the reliability of random fields and point images registration.

In Section 10.4.1, optimal search algorithms for point-pulse sources were described, which assumed the using of a single receiver with a tunable viewing window. In the presence of several receiving devices, the average localization time can be greatly reduced. The purpose of this paragraph 10.3.2 is to construct the time-optimal (in statistical terms) localization algorithms for randomly located point-pulse sources that take into account the number of receivers used and provide the required localization accuracy. Within this study, it is assumed that *a priori* information about the probable location of

a random pulse object inside of the search interval $(0, L)$ is absent, i.e. that the probability density of an unknown source on the x-axis is given by the function

$$f(x) = \begin{cases} \dfrac{1}{L}, & x \in (0, L) \\ 0, & x \notin (0, L). \end{cases}$$

The first question we need to answer is as follows: "What accuracy of localization can be achieved with a single-tact search procedure that is carried out with the help of n receiving devices and ends at the moment of first pulse generation by source?"

We pay special attention to the fact that, according to the conditions of the problem, the search procedure ends not at the moment of first pulse registration by the receiving system, but at the moment of random source's first pulse generation. This is not the same because in principle the receiving system can work in "missed-some-pulses" mode (not registering all the generated pulses). Thus, in single-tact search, it is required that the system doesn't miss any pulse. Obviously, in this case, the aggregate viewing window of all n receivers has to overlap the entire search interval $(0, L)$ at any moment. Let N be the number of elementary segments which the interval $(0, L)$ should be divided into (pulse object is located in one of such segment). Generally speaking, it would be possible, for example, to put N equal to the number of receivers n and, dividing the search interval into n equal parts, assign to each of $N = n$ segments their own available receiver. In such a search procedure, there is no problem to "link" the source to the desired segment, since the detected pulse is always fixed by only one receiver. But, unfortunately, such a simplified algorithm has an extremely low localization accuracy $\varepsilon = L/n$, and therefore it is very far from optimal.

(a) Optimal single-tact search procedure: The construction of a single-tact search procedure for n receiving devices, which really minimizes the localization error, is carried out as follows. Each of the n receivers is specified with the help of Table 10.3, the ith line of which describes the monitoring zone of the receiver with the number i. Binary variables x_{ij}, $i = \overline{1, n}$; $j = \overline{1, N}$ forming the table (note that the optimal value of the parameter N has yet to be determined)

Table 10.3. A two-dimensional array describing the observation zone for each of n receiving system's devices in a single-tact search procedure.

	Segments				
Receivers	1	2	3	$\cdots$	N
1th receiver	x_{11}	x_{12}	x_{13}		x_{1N}
2th receiver	x_{21}	x_{22}	x_{23}		x_{2N}
3th receiver	x_{31}	x_{32}	x_{33}		x_{3N}
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$
ith receiver	x_{i1}	x_{i2}	x_{i3}		x_{iN}
	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$
nth receiver	x_{n1}	x_{n2}	x_{n3}		x_{nN}

have the values 0 or 1 according to the following rule:

- if $x_{ij} = 0$, segment j does not enter the observation zone of receiver I,
- if $x_{ij} = 1$, the segment j enters the observation zone of the receiver i.

Thus, the row vector $\boldsymbol{x}_i = (x_{i1}, x_{i2}, \ldots, x_{iN})$ uniquely defines the receiver i, and the whole array (x_{ij}) uniquely fully describes the receiving system. The dynamic state of this system is characterized by the column vector $\boldsymbol{r} = (r_1, r_2, \ldots, r_n)^T$, in which all binary (i.e., possessing the value 0 or 1) variables r_i, $i = \overline{1,n}$ are equal to zero throughout the time interval from the beginning of the search to pulse's registration moment. When the pulse is fixed, each of the variables r_i goes (or not) into the state $r_i = 1$, depending on whether the received pulse was fixed by the ith receiver or not. The task is: by means of this changed column vector $\boldsymbol{r}$ to determine the number of the segment which has generated the pulse.

Three statements are formulated in the following that make the construction of an optimal single-tact search algorithm an almost obvious procedure:

Statement 1. In order for the single-stage search procedure to uniquely determine the segment number j, where it was the pulse-initiator, it is necessary and sufficient that all possible realizations of

the vector $\boldsymbol{r}$ characterizing the state of the receiving system at the moment of pulse fixation differ from each other.

Statement 2. The maximal number of elementary segments $N_{\max}$ initial search interval $(0, L)$ may be divided into (this number $N_{\max}$, actually sets the localization accuracy of the pulse source), is $N_{\max} = 2^n - 1$, where n is the number of receivers used (the total number of different states of the vector $\boldsymbol{r}$ is 2^n, but the system cannot be in the state $\boldsymbol{r} = \mathbf{0}$ when registering the pulse). If we try to exceed the number $N_{\max} = 2^n - 1$ it will become impossible to restore unambiguously the number of a segment with a pulse source.

Statement 3. It is expedient to carry out the creation of the binary table (x_{ij}) corresponding to the optimal algorithm for the single-tact random point source localization, not by combining row vectors $\boldsymbol{x}_i = (x_{i1}, x_{i2}, \ldots, x_{iN})$, $i = \overline{1, n}$, describing each of the n receivers, but by combining the column vector $\boldsymbol{x}_j^T = (x_{1j}, x_{2j}, \ldots, x_{nj})^T$, $j = \overline{1, N}$, characterizing the receiving system's dynamic response to the pulse in the segment j.

Considering statements 1–3, the optimal formation of the table (x_{ij}) is to choose one of $(2^n - 1)!$ variants to pack a two-dimensional array that includes all non-zero realization of column vector $\boldsymbol{x}_j^T = (x_{1j}, x_{2j}, \ldots, x_{nj})^T$. Each binary column $(x_{1j}, x_{2j}, \ldots, x_{nj})^T$ corresponds to its number $j = \sum_{i=1}^{n} r_i \times 2^{n-i}$ from the range $j = \overline{1, 2^n - 1}$.

Note that the order of the columns $\boldsymbol{x}_j^T$ in the formed array (x_{ij}) can be any. If the columns of the array follow in increasing order (by their "content"), then the matrix (x_{ij}) describing such a receiving system, is as simple as possible (see Table 10.4). In this case, the algorithm to find the segment number with an unknown source is very simple too: if the dynamic state of the receiving system is described by the vector $\boldsymbol{r} = (r_1, r_2, \ldots, r_n)^T$, then the pulse source is located in the segment with the number $j = \sum_{i=1}^{n} r_i \times 2^{n-i}$ and the set of scalar coordinates of the column vector $\boldsymbol{r}$ is the binary record of this number. For example, if the search is performed by three receivers ($n = 3$), then the surveillance zone should be divided into seven $(2^n - 1)$ segments, which are assigned to the receivers as follows: receiver №1 — segments 4,5,6,7; receiver №2 — segments 2,3,6,7; receiver №3 — segments 1,3,5,7. In this case, if the state vector of the system $\boldsymbol{r} = (0, 0, 1)^T$, i.e. if only the third receiver has

Table 10.4. A binary table corresponding to the receiving system with monotonically increasing numbers of pulse-initiator segments.

	Segments				
Receivers	1	2	3	$\cdots$	$N_{\max} = 2^n - 1$
1th receiver	0	0	0	$\cdots$	1
2th receiver	0	0	0	$\cdots$	1
3th receiver	0	0	0	$\cdots$	1
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$
ith receiver	0	0	0	$\cdots$	1
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$
$(n-1)$th receiver		1	1	$\cdots$	1
nth receiver	1	0	1	$\cdots$	1

worked, then the initiator of the pulse is located in the first segment; if $\boldsymbol{r} = (0,1,0)^T$, i.e. if only the second receiver has triggered, then the initiator of the pulse is the second segment; if $\boldsymbol{r} = (0,1,1)^T$, i.e. if the second and third receivers are triggered, the impulse initiator is the third segment, etc.

The absolute accuracy of the single-tact localization procedure of an unknown source is $\varepsilon = L/(2^n - 1)$, and the average localization time $\langle\tau\rangle$ is equal to the expected value of pause between the pulses

$$\langle\tau\rangle = \int_0^\infty t\mathrm{h(t)dt} = \int_0^\infty t\lambda \exp(-\lambda t)\mathrm{dt} = \frac{1}{\lambda}$$

and does not depend on the number of receivers used.

(b) Optimal multistep search using n receivers: The single-tact search procedure described above shows how the search effort should be distributed among the n receivers if it is required to reach the highest accuracy of random pulse source localization. The main moment which is directly used further is that the procedure described above is extremely constructive and completely unambiguous, so in the case of an optimal distribution of search efforts, the pulse object is always localized with accuracy $W/(2^n - 1)$, where W is the integrated viewing window uniting all n receivers used. Now, taking these considerations into account, we can begin to solve a more complicated task: it is necessary to construct a time-optimal algorithm when the

search for a random pulse source has to be carried out by a system of n receiving devices with the required localization accuracy ε. Except stipulated in advance localization accuracy, an important complicating factor in the formulated problem is that the choice of the optimum search procedure isn't limited only by single-tact localization algorithms, and there is no requirement to finish the search at the moment of the first pulse generation. Moreover, as is shown in the following, even at rather low requirements for the localization accuracy, the optimal algorithm is a multistage procedure (transition from one stage to another stage comes at the moment when the receiving system registers the next pulse). At the same time, it is quite acceptable that a part of the pulses generated by a random source is not fixed by the system. As for the previous problem, it is assumed that the localized point source has a uniform distribution density on the interval $(0, L)$.

Passing to the solution of the task, let us introduce a number of additional designations. The symbol M is the number of stages in the search procedure, and the symbol W_i is the aggregate system viewing window uniting n receivers at the ith stage of the search (in cases that do not allow double interpretation, the same symbol will mean not only the window itself, but also its linear dimension). Taking this into account, the average time $\langle\tau\rangle$ of any M-stage (not necessarily optimal) search procedure of a random source that guarantees the localization accuracy ε is

$$\langle\tau\rangle = \frac{1}{\lambda} \times \left(\frac{L}{W_1} + \frac{\frac{W_1}{2^n-1}}{W_2} + \frac{\frac{W_2}{2^n-1}}{W_3} + \cdots + \frac{\frac{W_{M-1}}{2^n-1}}{W_M} \right). \tag{10.54}$$

$$W_M = (2^n - 1)\varepsilon. \tag{10.54a}$$

The integer parameter M and continuous variables W_i, $i = \overline{1, M}$, which deliver the minimum to the expression (10.67), need to be estimated. It is taken into account in relation (10.54) that each subsequent $(i + 1)$th stage of the search procedure includes scanning by the cumulative viewing window W_{i+1} within one of the $2^n - 1$ segments that formed the viewing window W_i on the previous stage, and $(i + 1)$th scanning stage is carried out inside of the segment where ith pulse was fixed. For a fixed M, the optimal sizes of scanning viewing windows W_i, $i = \overline{1, M}$, for which average localization time (10.54)

reaches a minimum, have to satisfy the system of equations

$$\begin{cases} \dfrac{\partial\langle\tau\rangle}{\partial W_1} = \dfrac{1}{\lambda(2^n-1)} \times \left(-\dfrac{W_0}{W_1^2} + \dfrac{1}{W_2}\right) = 0; \\ \dfrac{\partial\langle\tau\rangle}{\partial W_2} = \dfrac{1}{\lambda(2^n-1)} \times \left(-\dfrac{W_1}{W_2^2} + \dfrac{1}{W_3}\right) = 0; \\ \qquad\vdots \\ \dfrac{\partial\langle\tau\rangle}{\partial W_{M-1}} = \dfrac{1}{\lambda(2^n-1)} \times \left(-\dfrac{W_{M-2}}{W_{M-1}^2} + \dfrac{1}{W_M}\right) = 0, \end{cases} \tag{10.55}$$

$$W_M = (2^n-1)\varepsilon, \tag{10.55a}$$

$$W_0 = (2^n-1)L. \tag{10.55b}$$

The system of equations (10.55) is obtained by simply equating to zero all the partial derivatives of average search time $\langle\tau\langle$ (10.54), and the relation (10.55b) is introduced for the symmetrization of the notation. It is more convenient to divide the further solution of the problem into two parts. First, we find the number M_{opt} of stages for the optimal localization procedure, and then we calculate all other parameters. Let us note that for a fixed value of the integer parameter M the solution of the system (10.55) is

$$W_i = \left[(2^n-1)(\varepsilon/L)^{\frac{i}{M}}\right] \times L, \quad i = \overline{1, M}. \tag{10.56}$$

Taking into account that the dimensionless coefficients $(2^n - 1)(\varepsilon/L)^{\frac{i}{M}}$ cannot be more than 1, we get the condition under which the solution (10.56) is valid:

$$(\varepsilon/L) \le \frac{1}{(2^n-1)^M}. \tag{10.56a}$$

Substituting (10.56) into (10.54), we obtain the expression for the average time $\langle\tau\rangle$ in the case of the M-stage localization procedure:

$$\langle\tau\rangle_M = \frac{1}{\lambda} \times \frac{M}{2^n-1}(\varepsilon/L)^{-\frac{1}{M}}. \tag{10.57}$$

Function

$$f(x) = \frac{x(\varepsilon/L)^{-\frac{1}{x}}}{\lambda(2^n-1)}, \tag{10.58}$$

which is a continuous analog of expression (10.57), has only one local minimum (this is used later), reached at the point $x_{\min} = -\ln(\varepsilon/L)$. Rewriting the condition (10.56a) in the equivalent form

$$M \le -\frac{\ln(\varepsilon/L)}{\ln(2^n - 1)}, \tag{10.59}$$

we find that for any value $n \ge 2$ (positive and integer), the number of M_{opt} stages in the optimal localization procedure is the same as the maximum integer satisfying the constraint (10.59). Here we use the fact that the function (10.58) more to the left of a point $x_{\min} = -\ln(\varepsilon/L)$ monotonously decreases. Thus, for the values ε/L in the neighborhood of the point

$$(\varepsilon/L) = \frac{1}{(2^n - 1)^M},$$

the optimal search procedure consists of M stages. For a complete description of the optimal localization algorithm, it remains to find out under what requirements the accuracy of localization (i.e. for what the values of ε/L) transition from M-stage to $(M + 1)$-stage search occurs. It is obvious that at the transition point, the average time of the M-stage search has to be equal to the average time of the most "fast" $(M + 1)$-stage search:

$$\frac{1}{\lambda} \times \frac{M}{2^n - 1}(\varepsilon/L)^{-\frac{1}{M}} = \frac{M+1}{\lambda}. \tag{10.60}$$

From here we obtain the point where the optimal localization algorithms have a transition from the M-stage search strategy to $(M + 1)$-stage one:

$$(\varepsilon/L)_{M \to M+1} = \frac{1}{(2^n - 1)^M} \times \left(\frac{M}{M+1}\right)^M. \tag{10.61}$$

The main calculation results that systematize the parameters of time-optimal localization algorithms of objects that in random time points generate instantaneous pulses (delta functions) are presented in Tables 10.5 and 10.6. These results are applicable to systems with several ($n \ge 2$) receivers. The last line of the table contains the parameters of the optimal search for the asymptotic procedure when the required localization accuracy ε/L tends to zero.

Table 10.5. Parameters of the optimal search for the random pulse source depending on the number of receivers $n(n \geq 2)$ and the required localization accuracy ε.

(ε/L) (required localization accuracy)	M^*_{opt}	$W_m, m = \overline{1, M_{\text{opt}}}$ (Viewing windows of the receiving system at each of M_{opt} stages of optimal search)	$\langle\tau\rangle$ (average localization time)
$\frac{1}{2^n-1} \leq (\varepsilon/L) < 1$	1	$W_1 = L$	$\frac{1}{\lambda}$
$\frac{1}{2(2^n-1)} \leq (\varepsilon/L) \leq \frac{1}{2^n-1}$	1	$W_1 = (2^n-1)(\varepsilon/L)^1 \times L$ $= (2^n-1)\varepsilon$	$\frac{1}{\lambda(2^n-1)}(\varepsilon/L)^{-1}$
$\frac{1}{(2^n-1)^2} \leq (\varepsilon/L) \leq \frac{1}{2(2^n-1)}$	2	$W_1 = (2^n-1)(\varepsilon/L)^{\frac{1}{2}} \times L$ $W_2 = (2^n-1)(\varepsilon/L) \times L$ $= (2^n-1)\varepsilon$	$\frac{2}{\lambda}$
$\frac{1}{(2^n-1)^2} \times \left(\frac{2}{3}\right)^2$ $\leq (\varepsilon/L) \leq \frac{1}{(2^n-1)^2}$	2	$W_1 = (2^n-1)(\varepsilon/L)^{\frac{1}{2}} \times L$ $W_2 = (2^n-1)(\varepsilon/L) \times L$ $= (2^n-1)\varepsilon$	$\frac{2}{\lambda(2^n-1)}(\varepsilon/L)^{-\frac{1}{2}}$
$\frac{1}{(2^n-1)^3} \leq (\varepsilon/L)$ $\leq \frac{1}{(2^n-1)^2} \times \left(\frac{2}{3}\right)^2$	3	$W_1 = L; \quad W_2 = \frac{1}{2^n-1} \times L;$ $W_3 = \frac{1}{(2^n-1)^2} \times L$	$\frac{3}{\lambda}$

$\frac{1}{(2^n-1)^3} \times \left(\frac{3}{4}\right)^3 \leq$ $\leq (\varepsilon/L) \leq \frac{1}{(2^n-1)^3}$	3	$W_1 = (2^n-1)(\varepsilon/L)^{\frac{1}{3}} \times L$ $W_2 = (2^n-1)(\varepsilon/L)^{\frac{2}{3}} \times L$ $W_3 = (2^n-1)(\varepsilon/L) \times L$ $= (2^n-1)\varepsilon$	$\frac{3}{\lambda(2^n-1)}(\varepsilon/L)^{-\frac{1}{3}}$
$\vdots$	$\vdots$	$\vdots$	$\vdots$
$\frac{1}{(2^n-1)^M} \leq (\varepsilon/L)$ $\leq \frac{1}{(2^n-1)^{M-1}} \left(\frac{M-1}{M}\right)^{M-1}$	M	$W_1 = L; \quad W_2 = \frac{1}{2^n-1} \times L$ $\ldots \quad W_m = \frac{1}{(2^n-1)^{m-1}} \times L$ $\ldots \quad W_M = \frac{1}{(2^n-1)^{M-1}} \times L$	$\frac{M}{\lambda}$
$\frac{1}{(2^n-1)^M} \left(\frac{M}{M+1}\right)^M$ $\leq (\varepsilon/L) \leq \frac{1}{(2^n-1)^M}$	M	$W_1 = (2^n-1)(\varepsilon/L)^{\frac{1}{M}} \times L$ $W_2 = (2^n-1)(\varepsilon/L)^{\frac{2}{M}} \times L$ $\ldots$ $W_m = (2^n-1)(\varepsilon/L)^{\frac{m}{M}} \times L$ $\ldots$ $W_M = (2^n-1)(\varepsilon/L)^{\frac{M}{M}} \times L$ $= (2^n-1)\varepsilon$	

(*Continued*)

Table 10.5. (*Continued*)

(ε/L) (required localization accuracy)	M^{*}_{opt}	$W_m, m = \overline{1, M_{\mathrm{opt}}}$ (viewing windows of the receiving system at each of M_{opt} stages of optimal search)	$\langle\tau\rangle$ (average localization time)
$\vdots$	$\vdots$	$\vdots$	$\vdots$
$(\varepsilon/L) \to 0$	M^{**}_{∞}	$W_1 = L; \quad W_2 = \dfrac{1}{2^n - 1} \times L$	$\dfrac{M_\infty}{\lambda} \approx$
$\dfrac{e^{-1}}{(2^n-1)^{M_\infty}} \le (\varepsilon/L)$		$\dots$	$\approx -\dfrac{\ln(\varepsilon/L)}{\lambda \ln(2^n-1)}$
$\le \dfrac{e^{-1}}{(2^n-1)^{M_\infty - 1}}$		$W_m = \dfrac{1}{(2^n-1)^{m-1}} \times L$	
		$\dots$	
		$W_{M_\infty} = \dfrac{1}{(2^n-1)^{M_\infty - 1}} \times L$	
		$\approx \dfrac{2^n - 1}{(2^n-1)^{\frac{-\ln(\varepsilon/L)}{\ln(2^n-1)}}} \times L$	
		$= (2^n - 1)\varepsilon$	

Notes: *Optimal number of stages for given localization accuracy. $^{**} M_\infty \approx \frac{-\ln(\varepsilon/L)}{\ln(2^n-1)}$.

Table 10.6. Optimal search parameters of a random pulse source for a system with one receiver ($n = 1$).

(ε/L) (required localization accuracy)	M^*_{opt}	$W_m, m = \overline{1, M_{\text{opt}}}$ (Viewing windows of the receiving system at each of M_{opt} stages of optimal search)	$\langle\tau\rangle$ (average localization time)
$\frac{1}{4} \le (\varepsilon/L) < 1$	1	$W_1 = \varepsilon$	$\frac{1}{\lambda}(\varepsilon/L)^{-1}$
$\left(\frac{2}{3}\right)^6 \le (\varepsilon/L) \le \frac{1}{4}$	2	$W_1 = (\varepsilon/L)^{\frac{1}{2}} \times L$ $W_2 = (\varepsilon/L) \times L = \varepsilon$	$\frac{2}{\lambda}(\varepsilon/L)^{-\frac{1}{2}}$
$\left(\frac{3}{4}\right)^{12} \le (\varepsilon/L) \le \left(\frac{2}{3}\right)^6$	3	$W_1 = (\varepsilon/L)^{\frac{1}{3}} \times L$ $W_2 = (\varepsilon/L)^{\frac{2}{3}} \times L$ $W_3 = (\varepsilon/L) \times L = \varepsilon$	$\frac{3}{\lambda}(\varepsilon/L)^{-\frac{1}{3}}$
$\vdots$	$\vdots$	$\vdots$	$\vdots$
$\left(\frac{M}{M+1}\right)^{M(M+1)}$ $\le (\varepsilon/L)$ $\le \left(\frac{M-1}{M}\right)^{M(M-1)}$	M	$W_1 = (\varepsilon/L)^{\frac{1}{M}} \times L$ $W_2 = (\varepsilon/L)^{\frac{2}{M}} \times L$ $\cdots$ $W_m = (\varepsilon/L)^{\frac{m}{M}} \times L$ $\cdots$ $W_M = (\varepsilon/L)^{\frac{M}{M}} \times L = \varepsilon$	$\frac{M}{\lambda}(\varepsilon/L)^{-\frac{1}{M}}$

(*Continued*)

Table 10.6. (*Continued*)

(ε/L) (required localization accuracy)	M^{*}_{opt}	$W_m, m = \overline{1, M_{\mathrm{opt}}}$ (Viewing windows of the receiving system at each of M_{opt} stages of optimal search)	$\langle\tau\rangle$ (average localization time)
$\vdots$	$\vdots$	$\vdots$	$\vdots$
			$\frac{M_\infty}{\lambda}(\varepsilon/L)^{-\frac{1}{M_\infty}} \approx$
$(\varepsilon/L) \to 0$			$\approx \frac{-\ln(\varepsilon/L)}{\lambda}\times$
$(\varepsilon/L) \approx e^{-M_\infty}$	M^{**}_∞	$W_1 = (\varepsilon/L)^{\frac{1}{M_\infty}} \times L = e^{-1} \times L$	$\times(\varepsilon/L)^{\frac{1}{\ln(\varepsilon/L)}} =$
		$W_2 = (\varepsilon/L)^{\frac{2}{M_\infty}} \times L = e^{-2} \times L$	$= -\frac{e\ln(\varepsilon/L)}{\lambda}$
		$\dots$	
		$W_m = (\varepsilon/L)^{\frac{m}{M_\infty}} \times L = e^{-m} \times L$	
		$\dots$	
		$W_{M_\infty} = e^{-M_\infty} \times L = (\varepsilon/L) \times L = \varepsilon$	

Notes: *Optimal number of stages for given localization accuracy. **$M_\infty \approx -\ln(\varepsilon/L)$.

The results solve the problem of creating the algorithms of time-optimal localization of random pulse point sources when such objects have a uniform distribution on the search interval $(0, L)$. The proposed search strategies open up the prospect of finding the optimal localization algorithms in cases when the probability density function of the random pulse source differs from uniform. Another interesting and poorly investigated direction of the problem is the construction of optimal search procedures focused on the simultaneous localization of several random sources.

The methods directed at constructing the algorithms of the optimal search for random pulse point objects, presented here, were published in [19, 21–24, 29].

10.5 Conclusion

In the introduction, we have already noted that many questions connected with or reducing to problems with a random partition of an interval are simple in the formulation, but are very difficult to solve. It is appropriate here to recall the great mathematician R. Bellman. On the question of how to distinguish a mathematical problem from a usual exercise, he answered approximately so: "In my opinion, everything is very simple. I'm trying to solve the problem. If it manages, this is an exercise. If not, it's the problem". The evidential constructions and methods developed in this chapter confirm this idea and show that finding even a particular solution to a simple (in formulation) probability task very often leads to a serious scientific problem. For example, in the proof of formula (10.14), we had to use the help of a computer with "intellectual filling", and only with its help "the hypothesis" at first has arisen and many years later has been proved. Moreover — in the proof of the relation (10.33) — even powerful program support has appeared insufficiently. As a result, in order to find and prove relation (10.33), we had to propose a new concept of generalized three-dimensional Catalan numbers (and to find their explicit expressions) and then to solve some rather difficult combinatorial probability problems.

So, we have demonstrated the example of effective computer use for intellectual software support, required for difficult probability tasks solving. There are not so many examples of such non-standard

use of computers in scientific practice. The brightest of these is, apparently, the proof of four color theorem. However, in our case, the computer was not only a powerful "calculator" but also an "intelligent helper" having tools for dealing with branched and time-consuming analytical calculations.

Our successes in constructing the analytical software manipulation systems and the experience gained in the process of their creation allow us to hope that soon it is possible to enlarge the list of successful computer applications by the problematic scientific problems already solved.

Acknowledgments

The work is supported by Russian Science Foundation, grant No. 24-21-00136 and Ministry of Science and Higher Education of the Russian Federation (project number 124041700103-1).

Appendix

Table A.1. Computer-calculated formulas for the probabilities $P_{n,k}(\varepsilon)$.

N	k	ε boundaries	$P_{n,k}(\varepsilon)$
3	2	(0, 1)	$1-3\varepsilon^2+2\varepsilon^3$
4	2	(0, 1/2)	$1-12\varepsilon^2+24\varepsilon^3-14\varepsilon^4$
		(1/2, 1)	$2-8\varepsilon+12\varepsilon^2-8\varepsilon^3+2\varepsilon^4$
	3	(0, 1)	$1-4\varepsilon^3+3\varepsilon^4$
5	2	(0, 1/2)	$1-30\varepsilon^2+100\varepsilon^3-120\varepsilon^4+48\varepsilon^5$
	3	(0, 1/2)	$1-20\varepsilon^3+40\varepsilon^4-22\varepsilon^5$
		(1/2, 1)	$10\varepsilon-40\varepsilon^2+60\varepsilon^3-40\varepsilon^4+10\varepsilon^5$
	4	(0, 1)	$1-5\varepsilon^4+4\varepsilon^5$
6	2	(0, 1/3)	$1-60\varepsilon^2+280\varepsilon^3-420\varepsilon^4+12\varepsilon^5+320\varepsilon^6$
		(1/3, 1/2)	$5-60\varepsilon+300\varepsilon^2-800\varepsilon^3+1200\varepsilon^4-960\varepsilon^5+320\varepsilon^6$
	3	(0, 1/2)	$1-60\varepsilon^3+195\varepsilon^4-222\varepsilon^5+85\varepsilon^6$
		(1/2, 1)	$5-30\varepsilon+75\varepsilon^2-100\varepsilon^3+75\varepsilon^4-30\varepsilon^5+5\varepsilon^6$
	4	(0, 1/2)	$1-30\varepsilon^4+60\varepsilon^5-32\varepsilon^6$
		(1/2, 1)	$2-12\varepsilon+60\varepsilon^2-160\varepsilon^3+210\varepsilon^4-132\varepsilon^5+32\varepsilon^6$
	5	(0, 1)	$1-6\varepsilon^5+5\varepsilon^6$

(*Continued*)

Table A.1. (*Continued*)

N	k	ε boundaries	$P_{n,k}(\varepsilon)$
7	2	(0, 1/4)	$1-105\varepsilon^2+630\varepsilon^3-910\varepsilon^4-2604\varepsilon^5+9583\varepsilon^6-8446\varepsilon^7$
		(1/4, 1/3)	$28\varepsilon-441\varepsilon^2+2870\varepsilon^3-9870\varepsilon^4+18900\varepsilon^5-19089\varepsilon^6+7938\varepsilon^7$
	3	(0, 1/2)	$1-140\varepsilon^3+630\varepsilon^4-1092\varepsilon^5+840\varepsilon^6-240\varepsilon^7$
	4	(0, 1/2)	$1-105\varepsilon^4+336\varepsilon^5-371\varepsilon^6+138\varepsilon^7$
		(1/2, 1)	$-7+84\varepsilon-357\varepsilon^2+770\varepsilon^3-945\varepsilon^4+672\varepsilon^5-259\varepsilon^6+42\varepsilon^7$
	5	(0, 1/2)	$1-42\varepsilon^5+84\varepsilon^6-44\varepsilon^7$
		(1/2, 1)	$14\varepsilon-84\varepsilon^2+280\varepsilon^3-560\varepsilon^4+630\varepsilon^5-364\varepsilon^6+84\varepsilon^7$
	6	(0, 1)	$1-7\varepsilon^6+6\varepsilon^7$
8	2	(0, 1/4)	$1-168\varepsilon^2+1232\varepsilon^3-1260\varepsilon^4-18480\varepsilon^5+85064\varepsilon^6-146640\varepsilon^7+91854\varepsilon^8$
		(1/4, 1/3)	$14-336\varepsilon+3528\varepsilon^2-21168\varepsilon^3+79380\varepsilon^4-190512\varepsilon^5+285768\varepsilon^6-244944\varepsilon^7+91854\varepsilon^8$
	3	(0, 1/3)	$1-280\varepsilon^3+1610\varepsilon^4-3752\varepsilon^5+4900\varepsilon^6-4952\varepsilon^7+3311\varepsilon^8$
		(1/3, 1/2)	$-14+280\varepsilon-2212\varepsilon^2+9296\varepsilon^3-22960\varepsilon^4+34048\varepsilon^5-29120\varepsilon^6+12544\varepsilon^7-1792\varepsilon^8$
	4	(0, 1/2)	$1-280\varepsilon^4+1232\varepsilon^5-2072\varepsilon^6+1552\varepsilon^7-434\varepsilon^8$
		(1/2, 1)	$14-112\varepsilon+392\varepsilon^2-784\varepsilon^3+980\varepsilon^4-784\varepsilon^5+392\varepsilon^6-112\varepsilon^7+14\varepsilon^8$
	5	(0, 1/2)	$1-168\varepsilon^5+532\varepsilon^6-576\varepsilon^7+210\varepsilon^{89}$
		(1/2, 1)	$14-168\varepsilon+924\varepsilon^2-2800\varepsilon^3+5040\varepsilon^4-5544\varepsilon^5+3668\varepsilon^6-1344\varepsilon^7+210\varepsilon^8$
	6	(0, 1/2)	$1-56\varepsilon^6+112\varepsilon^7-58\varepsilon^8$
		(1/2, 1)	$2-16\varepsilon+112\varepsilon^2-448\varepsilon^3+1120\varepsilon^4-1792\varepsilon^5+1736\varepsilon^6-912\varepsilon^7+198\varepsilon^8$
	7	(0, 1)	$1-8\varepsilon^7+7\varepsilon^8$
9	2	(0, 1/5)	$1-252\varepsilon^2+2184\varepsilon^3-504\varepsilon^4-77616\varepsilon^5+439908\varepsilon^6-1062072\varepsilon^7+\quad+1112679\varepsilon^8-304904\varepsilon^9$
		(1/4, 1/5)	$-6+288\varepsilon-5472\varepsilon^2+56784\varepsilon^3-362754\varepsilon^4+1497384\varepsilon^5-4022592\varepsilon^6+6812928\varepsilon^7-6621696\varepsilon^8+2820096\varepsilon^9$
	3	(1/2, 1/3)	$42-756\varepsilon+6048\varepsilon^2-28224\varepsilon^3+84672\varepsilon^4-169344\varepsilon^5+225792\varepsilon^6-193536\varepsilon^7+96768\varepsilon^8-21504\varepsilon^9$
		(1/4, 1/3)	$36\varepsilon-576\varepsilon^2+4872\varepsilon^3-28728\varepsilon^4+118692\varepsilon^5-323064\varepsilon^6+541296\varepsilon^7-502470\varepsilon^8+197196\varepsilon^9$
		(0, 1/4)	$1-504\varepsilon^3+3528\varepsilon^4-10332\varepsilon^5+21000\varepsilon^6-48528\varepsilon^7+87354\varepsilon^8-64948\varepsilon^9$
	4	(0, 1/2)	$1-630\varepsilon^4+3528\varepsilon^5-7980\varepsilon^6+9000\varepsilon^7-5040\varepsilon^8+1120\varepsilon^9$
	5	(0, 1/2)	$1-504\varepsilon^5+2184\varepsilon^6-3600\varepsilon^7+2646\varepsilon^8-728\varepsilon^9$
		(1/2, 1)	$-42+504\varepsilon-2520\varepsilon^2+7056\varepsilon^3-12348\varepsilon^4+14112\varepsilon^5-10584\varepsilon^6+5040\varepsilon^7-1386\varepsilon^8+168\varepsilon^9$

(*Continued*)

Table A.1. (*Continued*)

N	k	ε boundaries	$P_{n,k}(\varepsilon)$
	6	(0, 1/2)	$1 - 256\varepsilon^6 + 792\varepsilon^7 - 846\varepsilon^8 + 304\varepsilon^9$
		(1/2, 1)	$-18 + 288\varepsilon - 1908\varepsilon^2 + 7224\varepsilon^3 - 17136\varepsilon^4 + 28208\varepsilon^5 - 25788\varepsilon^6 + 15768\varepsilon^7 - 5454\varepsilon^8 + 816\varepsilon^9$
	7	(0, 1/2)	$1 - 72\varepsilon^7 + 144\varepsilon^8 - 74\varepsilon^9$
		(1/2, 1)	$18\varepsilon - 144\varepsilon^2 + 672\varepsilon^3 - 2016\varepsilon^4 + 4032\varepsilon^5 - 5376\varepsilon^6 + 4536\varepsilon^7 - 2160\varepsilon^8 + 438\varepsilon^9$
	8	(0, 1)	$1 - 9\varepsilon^8 + 8\varepsilon^9$
10	2	(0, 1/6)	$1 - 360\varepsilon^2 + 3600\varepsilon^3 + 3360\varepsilon^4 - 247464\varepsilon^5 + 1653960\varepsilon^6 - 4733040\varepsilon^7 + 4405320\varepsilon^8 + 5988980\varepsilon^9 - 11738484\varepsilon^{10}$
		(1/6, 1/5)	$2 - 60\varepsilon + 1260\varepsilon^2 - 22320\varepsilon^3 + 275520\varepsilon^4 - 2207016\varepsilon^5 + 11451720\varepsilon^6 - 38325360\varepsilon^7 + 79988040\varepsilon^8 - 94787980\varepsilon^9 + 48727692\varepsilon^{10}$
		(1/5, 1/4)	$42 - 1680\varepsilon + 30240\varepsilon^2 - 322560\varepsilon^3 + 2257920\varepsilon^4 - 10838016\varepsilon^5 + 36126720\varepsilon^6 - 82575360\varepsilon^7 + 123863040\varepsilon^8 - 110100480\varepsilon^9 + 44040192\varepsilon^{10}$
11	2	(0, 1/6)	$1 - 495\varepsilon^2 + 5610\varepsilon^3 + 13860\varepsilon^4 - 659736\varepsilon^5 + 5019630\varepsilon^6 - 15057900\varepsilon^7 - 4097115\varepsilon^8 + 151449540\varepsilon^9 - 383928336\varepsilon^{10} + 322511040\varepsilon^{11}$
		(1/6, 1/5)	$-33 + 1980\varepsilon - 52470\varepsilon^2 + 816420\varepsilon^3 - 8325900\varepsilon^4 + 58616712\varepsilon^5 - 291362610\varepsilon^6 + 1024204500\varepsilon^7 - 2498326875\varepsilon^8 + 4031362500\varepsilon^9 - 3875850000\varepsilon^{10} + 1683000000\varepsilon^{11}$
12	2	(0, 1/7)	$1 - 660\varepsilon^2 + 8360\varepsilon^3 + 36630\varepsilon^4 - 1547568\varepsilon^5 + 13041336\varepsilon^6 - 36072432\varepsilon^7 - 139347450\varepsilon^8 + 1428925960\varepsilon^9 - 4725032796\varepsilon^{10} + 7241178696\varepsilon^{11} - 4179927862\varepsilon^{12}$
		(1/7, 1/6)	$11 - 792\varepsilon + 27984\varepsilon^2 - 616880\varepsilon^3 + 9205020\varepsilon^4 - 96627168\varepsilon^5 + 727406064\varepsilon^6 - 3949548768\varepsilon^7 + 15351496380\varepsilon^8 - 41691785520\varepsilon^9 + 75175109064\varepsilon^{10} - 80891098992\varepsilon^{11} + 39321260484\varepsilon^{12}$
		(1/6, 1/5)	$132 - 7920\varepsilon + 217800\varepsilon^2 - 3630000\varepsilon^3 + 40837500\varepsilon^4 - 326700000\varepsilon^5 + 1905750000\varepsilon^6 - 8167500000\varepsilon^7 + +25523437500\varepsilon^8 - 56718750000\varepsilon^9 + 85078125000\varepsilon^{10} - 77343750000\varepsilon^{11} + 32226562500\varepsilon^{12}$
13	2	(0, 1/8)	$1 - 858\varepsilon^2 + 12012\varepsilon^3 + 80080\varepsilon^4 - 3294720\varepsilon^5 + 30086628\varepsilon^6 - 62925720\varepsilon^7 - 909369747\varepsilon^8 + 607341600\varepsilon^9 - 34250637564\varepsilon^{10} + 8277827280\varepsilon^{11} - 54680029443\varepsilon^{12} - 3365355508\varepsilon^{13}$

(*Continued*)

Table A.1. (*Continued*)

N	k	ε boundaries	$P_{n,k}(\varepsilon)$
		(1/8, 1/7)	$104\varepsilon - 5850\varepsilon^2 + 158444\varepsilon^3 - 2848560\varepsilon^4 + 38877696\varepsilon^5 - 419752476\varepsilon^6 + 3535787112\varepsilon^7 - 22501646739\varepsilon^8 + 104573017120\varepsilon^9 - 341340799228\varepsilon^{10} + 738292725456\varepsilon^{11} - 948033227011\varepsilon^{12} + 546390458380\varepsilon^{13}$
		(1/7, 1/6)	$-143+11440\varepsilon-417846\varepsilon^2+9241804\varepsilon^3-138278855\varepsilon^4+1479488868\varepsilon^5 - 11654720220\varepsilon^6 + 68485295736\varepsilon^7 - 300346819344\varepsilon^8 + 971335553760\varepsilon^9 - 2252457979200\varepsilon^{10} + 3547733904000\varepsilon^{11} - 3402622080000\varepsilon^{12} + 1501156800000\varepsilon^{13}$
14	2	(0, 1/8)	$1 - 1092\varepsilon^2 + 16744\varepsilon^3 + 156156\varepsilon^4 - -6498492\varepsilon^5 + 63159096\varepsilon^6 - 56566224\varepsilon^7 - 3999887892\varepsilon^8 + 38992765812\varepsilon^9 - 174352258080\varepsilon^{10} + 360697473864\varepsilon^{11} - 4817094828\varepsilon^{12} - 1349321055012\varepsilon^{13} + 1753807624920\varepsilon^{14}$
		(1/8, 1/7)	$65 - 6552\varepsilon + 309036\varepsilon^2 - 8975512\varepsilon^3 + 178510332\varepsilon^4 - 2564958396\varepsilon^5 + 27418999608\varepsilon^6 - 221377405392\varepsilon^7 + 1356313562604\varepsilon^8 - 6275548683404\varepsilon^9 + 21590662949856\varepsilon^{10} - 53575501829304\varepsilon^{11} + 90670532458324\varepsilon^{12} - 93708297788196\varepsilon^{13} + 44634761108184\varepsilon^{14}$
		(1/7, 1/6)	$429 - 36036\varepsilon + 1405404\varepsilon^2 - 33729696\varepsilon^3 + 556539984\varepsilon^4 - 6678479808\varepsilon^5 + 60106318272\varepsilon^6 - 412157611008\varepsilon^7 + 2163827457792\varepsilon^8 - 8655309831168\varepsilon^9 + 25965929493504\varepsilon^{10} - 56652937076736\varepsilon^{11} - 84979405615104\varepsilon^{12} - 78442528260096\varepsilon^{13} + 3618226397184\varepsilon^{14}$

References

[1] Andre D. (1887) Solution directe du probleme resola par M. Bertrand. *Comptes Rendus de l'Académie des Sciences* (Paris), 105, 436–437.

[2] Barton, D.E. and David, F.N. (1959) Combinatorial extreme value distributions. *Mathematika*, 6, 63–76.

[3] Bezhanova M.M., Katkov V.L., and Pottosin I.V. (1972) Work on analytical transformations in the Computing Center of the SB RAS

USSR. *Computational Mathematics and Computer Science*, Issue 3. ILTPE of the Academy of Sciences of the Ukrainian SSR, Kharkov (in Russian).

[4] Darling, D.A. (1953) On class problems related to the random division of an interval. *Annals of Mathematical Statistics*, 24, 239–253.

[5] David, G. (1979) *Ordinal Statistics* (Nauka, Moscow) (in Russian).

[6] Efimov, V.M., Iskoldsky, A.M., Livshits, Z.A., and Krendel, Y.M. (1973) On the characteristics of various methods for discrete structure images readout. *Avtometriya*, 1, 3–7 (in Russian).

[7] Efimov, V.M. and Reznik, A.L. (1976) Analytical computer computation of volumes bounded by a system of hyperplanes in n-dimensional space. *Avtometriya*, 1, 116–119 (in Russian).

[8] Efimov, V.M. and Reznik, A.L. (1977) Analytical determination with the help of a computer of the statistical characteristics of the aperture scanning process of the Bernoulli flow. *Avtometriya*, 4, 49–51 (in Russian).

[9] Efimov, V.M., Nesterov, A.A., and Reznik, A.L. (1980) Algorithms of time-optimal search for light point objects. *Avtometriya*, 3, 72–76 (in Russian).

[10] Efimov, V.M., Zolotukhin, Y.N., and Reznik, A.L. (1991) Asymptotically optimal decorrelation of a stationary sequence of signal equidistant pixels. *Optoelectronics, Instrumentation and Data Processing*, 5, 98–101.

[11] Feller, W. (1968) *An Introduction to Probability Theory and Its Applications,* Vol. 1, 3rd Ed. (Wiley, New York).

[12] Fulton, W. (2006) Young Tables and Their Applications to the Theory of Representations and Geometry (MCCME, Moscow) (in Russian).

[13] Gerdt, V.P., Tarasov, O.V., and Shirkov, D.V. (1980) Analytical calculations on computers in the application to physics and mathematics. *Uspekhi Fizicheskih Nauk*, 130(1), 113–147 (in Russian).

[14] Gessel, I. and Zeilberger, D. (1992) Random walk in a Weyl chamber. *Proceedings of the American Mathematical Society*, 115(1), 27–31.

[15] Glushkov, V.M., Grinchenko, T.A., Dorodnicyna A.A., *et al.* (1978) ANALYST-74. *Cybernetics*, 5, 114–147 (in Russian).

[16] Kantorovich, L.V. (1957) About numerical and analytical calculations on computers with programmed control. *Series of Physics and Mathematics* (USSR Academy of Sciences), 10, 2 (in Russian).

[17] Naus, J.I. (1966) Some probabilities, expectations, and variances for the size of largest clusters and smallest intervals. *Journal of the American Statistical Association*, 61, 1191–1199.

[18] Parzen, E. (1960) *Modern Probability Theory and Its Applications* (John Wiley and Sons Inc., New York).

[19] Reznik, A. L. (1980) On intensity estimation of streams at registration of signal samples. *Avtometriya*, 3, 123–126 (in Russian).

[20] Reznik, A.L. (1981) Computer simulation of continuous readout of discrete structure images. *Avtometriya*, 6, 3–6 (in Russian).

[21] Reznik, A.L. (1991) The programs for analytical calculations in the problems of point object localization. *Optoelectronics, Instrumentation and Data Processing*, 6, 21–24.

[22] Reznik, A.L., Soloviev, A.A., and Torgov, A.V. (2018) The algorithms of optimum localization of random pulsed-point source under the condition of its uniform distribution on a search interval. *Pattern Recognition and Image Analysis. Advances in Mathematical Theory and Applications*, 28(2), 354–361.

[23] Reznik, A.L., Tuzikov, A.V., Soloviev, A.A., Torgov, A.V., and Kovalev, V.A. (2019) Time-optimal algorithms focused on the search for random pulsed-point sources. *Computer Optics*, 43(4), 605–610.

[24] Reznik, A.L., Soloviev, A.A., and Torgov, A.V. (2020) Localization of random pulse point sources using physically implementable search algorithms. *Optoelectronics, Instrumentation and Data Processing*, 56(6), 598–607.

[25] Reznik, A.L. and Soloviev, A.A. (2022) Software and combinatorial-probabilistic tools for the analysis of random point structures. *Pattern Recognition and Image Analysis*, 32(3), 636–638.

[26] Reznik, A.L., Tuzikov, A.V., Soloviev, A.A., and Torgov, A.V. (2023) Symbolic computation programs and generalized Catalan numbers in problems of analysis of random point structures. *Computational Technologies*, 28(1), 61–71.

[27] Reznik, A.L., Efimov, V.M., and Soloviev, A.A. (2011) Computer-analytical calculation of the probability characteristics of readout of random point images. *Optoelectronics, Instrumentation and Data Processing*, 47(1), 7–11.

[28] Reznik, A.L., Soloviev, A.A., and Torgov, A.V. (2016) Program-combinatorial approach to solving problems of error-free readout of random point images. *Optoelectronics, Instrumentation and Data Processing*, 52(2), 121–127.

[29] Reznik, A.L., Tuzikov, A.V., Soloviev, A.A., and Torgov, A.V. (2017) Time-optimal algorithms of searching for pulsed-point sources for systems with several detectors. *Optoelectronics, Instrumentation and Data Processing*, 53(3), 203–209.

[30] Shurygin, V.A. and Yanenko, N.N. (1961) On the realization of algebraic-differential algorithms on the computer. *Problems of Cybernetics*, 6 (Fizmatgiz, Moscow) (in Russian).
[31] Wilks, S. (1967) *Mathematical Statistics* (Nauka, Moscow) (in Russian).

© 2025 World Scientific Publishing Company
https://doi.org/10.1142/9789811267215_0011

Chapter 11

Collective Generalized Solutions for Infeasible Systems of Constraints and Ensemble Learning Techniques

Vladimir D. Mazurov[*] **and Mikhail Yu. Khachay**[†,‡]

[*]*Ural Federal University, Ekaterinburg, the Russian Federation*
[†]*Krasovsky Institute of Mathematics and Mechanics, Ekaterinburg, Russia*
[‡]*mkhachay@imm.uran.ru*

This chapter proposes a short survey of the recent results concerning the majoritarian generalized solutions for infeasible systems of algebraic constraints. In the literature, this kind of generalized solutions is also referred to as committee solutions or just committees. The survey deals with the existence theorems for such solutions, related game theoretic and combinatorial optimization problems, and ensemble learning algorithms.

Keywords: Infeasible constraint system, committee generalized solution, majoritary algorithms of ensemble learning, affine separating committees.

11.1 Introduction

A system of constraints (algebraic or differential equations, inequalities, their combinations, and more abstract inclusions) is a mathematical model that is widely involved in operations research, decision-making, machine learning, and data mining. In many cases, the investigated practical problem includes contradictory restrictions, which leads to inconsistency of the resulting model. Any time, when this happens, the considered system of constraints has no solutions in the common sense. To this end, before solving the system, it is required previously to formalize the concept of a solution itself.

In machine learning, there are known two celebrated approaches to perform such a generalization. The former is the famous *kernel trick* (see, e.g. [8, 17, 51]) designed to (implicitly) transform infeasible quadratic programs arising the in training of SVM classifier on inseparable samples to appropriate feature spaces. The latter approach concerns with developing *ensemble learning techniques*, among them are algebraic methods over families of weak classifiers (see, e.g. [9, 47, 53–56]), bagging [6, 7], boosting [12, 49, 50], etc.

In optimization, along with the famous [8, 10, 11] Chebyshev ε-relaxation continuous approach to the optimal correction of infeasible systems, an alternative discrete approach to such a generalization is actively developed [24, 35, 39–42]. According to this approach, the concept of a solution for a system of constraints (in case of its infeasibility) is refined using the concept of a finite ensemble of appropriately chosen quasi-solutions. The theory of *committee generalized solutions* based on fundamental results of Mazurov [39] provides a mathematical framework for this generalization, where partial quasi-solutions, also known as *committee members*, are combined by the majority voting.

The notion of a committee solution was introduced in the middle of 1960s [1, 44]. Early papers mostly deal with simple examples illustrating the applicability of committees to generalizing the solution concept and design of first learning heuristics. To the best of our knowledge, the first theoretic result in this field was Mazurov's theorem [38] yielding a criterion of perfect learnability in the class of affine separating committees. Further, the existencial theory for committee solutions was developed in numerous papers including [18, 20, 21, 30, 39, 45, 48]. Issues concerning the minimum committee solutions for a system of linear inequalities and closely

related piecewise-linear separation problems were investigated in [22–25, 31, 33, 40, 41, 43], respectively. More recent results in the field of efficient committee construction techniques and related ensemble learning algorithms are presented in [26–29, 32, 34, 36].

This chapter is structured as follows. In Section 11.2, we introduce a concept of the committee generalized solution for an abstract infeasible system of constraints. Section 11.3 is devoted to game-theoretic aspects related to the existence of committee solutions. In Section 11.4, we overview the recent results obtained for committee solutions of systems of linear inequalities and the Minimal Affine Separating Committee (MASC) combinatorial optimization problem related to ensemble learning techniques over the family of linear classifiers.

11.2 Committee Solutions: Basic Concepts

In this section, we introduce the main concept of a committee generalized solution and discuss some its basic properties.

Suppose, we are given by a ground set X and a family of its non-empty subsets $D_1, D_2, \ldots, D_m$. Consider the system of abstract inclusions

$$x \in D_j \quad (j \in [m]), \tag{11.1}$$

where $[m]$ is the integer segment $\{1, 2, \ldots, m\}$. If $\bigcap_{j=1}^{m} D_j = \varnothing$, system (11.1) is called *infeasible*. Any non-empty subset $L \subset [m]$ induces a *subsystem* of system (11.1). Without loss of generality, we do not distinguish a subset L and the appropriate subsystem. If $D(L) = \bigcap_{j \in L} D_j \neq \varnothing$, the subsystem is called feasible. Any maximal by inclusion subsystem of system (11.1) is called its *maximal feasible subsystem* or m.f.s.

Definition 11.1. A finite sequence $Q = (x^1, \ldots, x^q)$, $x^i \in X$, such that, for any $j \in [m]$, $|\{i\colon x^i \in D_j\}| > q/2$, is called a committee generalized solution (committee) of system (11.1).

The number q is called *a length* of the committee Q, and we say that system (11.1) has a committee solution (or solvable by a committee) of length q. A committee of minimal length q (for a given system (11.1)) is called its *minimum* committee solution or just a minimum committee.

It is convenient to represent the set of all committee solutions of the system (11.1) as follows. As in [39], introduce the vector function

$$\varphi\colon X \to \{-1,1\}^m, \quad \text{where } \varphi_j(x) = \begin{cases} 1, & \text{if } x \in D_j, \\ -1, & \text{otherwise.} \end{cases}$$

By construction, the image $\varphi(X)$ is a finite set. Let $\varphi(X) = \{\varphi^1, \ldots, \varphi^s\}$. Without loss of generality, we can assume that, the vectors φ^i are incomparable, i.e. for any i_1 and i_2, the equation $\varphi^{i_1} \geq \varphi^{i_2}$ implies $i_1 = i_2$. This means that any φ^i is a characteristic vector of some m.f.s. of system (11.1).

As it follows from Definition 11.1, a finite sequence Q is a committee solution of (11.1) in and only if, by some permutation, Q can be represented in the form

$$(\underbrace{y^{1,1}, \ldots, y^{1,z_1}}_{z_1}, \ldots, \underbrace{y^{s,1}, \ldots, y^{s,z_s}}_{z_s}), \tag{11.2}$$

where $\varphi(y^{i,l}) = \varphi^i$ and $z_1, \ldots, z_s$ are non-negative integers, such that $\sum_{i=1}^{s} z_i \varphi^i \geq e = [1,1,\ldots,1]^T$. It can easily verified that a sequence Q is a minimum committee of system (11.1) if and only if the vector z is a optimal solution in the following integer linear program:

$$\min \left\{ \sum_{i=1}^{s} z_i \colon \sum_{i=1}^{s} z_i \varphi^i \geq e, \ z \in \mathbb{Z}_+^s \right\}.$$

Obviously, if system (11.1) is feasible, its minimum committees are of length 1 and coincide with regular solutions. Otherwise, any minimum committee has more than 1 entry.

It is convenient to represent existence conditions for committee solutions of system (11.1) in terms of graph and hypergraph of its m.f.s.

Definition 11.2. A finite graph $G = (V, E)$ is called an m.f.s. graph of system (11.1) if its nodeset V consists of index sets $J_1, J_2, \ldots, J_s$ of this system, and $\{J_i, J_j\} \in E \iff J_i \cup J_j = [m]$.

The notion of an m.f.s. graph was initially coined in [52] for the case of linear inequalities (see, also [14]). Historically, the interest to m.f.s. graphs was motivated by the following simple sufficient condition.

Assertion 11.1. Let s be a natural number, $J_1, \ldots, J_{2s-1}$ be a cycle in the m.f.s. graph of system (11.1), and $x^i \in D(J_i)$. Then, the sequence $Q = (x^1, \ldots, x^{2s-1})$ is a committee solution of system (11.1).

To obtain necessary and sufficient conditions, we need to introduce more general notion (see, e.g. [19]).

Definition 11.3. A finite hypergraph $G = (V, E)$, whose nodeset V coincides with the family $\{J_1, \ldots, J_s\}$ of index sets of the m.f.s. of system (11.1) such that

$$\{J_{i_1}, \ldots, J_{i_t}\} \in E \iff \bigcup_{k=1}^{t} = [m],$$

is called an m.f.s. hypergraph of system (11.1).

The following theorem describes conditions for a finite hypergraph to be an m.f.s. hypergraph of some system (11.1) (up to an isomorphism). As usual, we say that hypergraphs $G = (V, E)$ and $\Gamma = (V\Gamma, E\Gamma)$ are isomorphic if there is a bijection $\psi : V\Gamma \to V$, such that

$$\{v_1, \ldots, v_p\} \in V\Gamma \iff \{\psi(v_1), \ldots, \psi(v_p)\} \in E.$$

Theorem 11.1. *Let $\Gamma = (V\Gamma, E\Gamma)$ be a non-empty*[1] *finite hypergraph without multiple edges. Γ is isomorphic to an m.f.c. hypergraph $G = (V, E)$ of an inclusion system*

$$x \in D_j(\Gamma) \quad (j \in [m]) \tag{11.3}$$

for some $m = m(\Gamma)$ if and only if Γ satisfies the following conditions:

$$\text{if } |V\Gamma| > 1, \text{ then } E\Gamma \text{ has no loops} \tag{11.4}$$

$$(u \in E\Gamma, u \subset w) \Rightarrow w \in E\Gamma. \tag{11.5}$$

Proof. *Necessity*: Let the hypergraph Γ specified in the hypothesis of Theorem 11.1 be isomorphic to the m.f.s. hypergraph $G = (V, E)$ of system (11.3) and let ψ: $V\Gamma \to V$ be such an isomorphism. Let us prove that conditions (11.4) and (11.5) hold.

[1] That is, $E\Gamma \neq \varnothing$.

Indeed, suppose $s = |V\Gamma| = |V| > 1$. Then, system (11.3) is infeasible and its m.f.s. hypergraph G contains no loops. Therefore, $E\Gamma$ has no loops as well. Further, consider an arbitrary hyperedge $u = \{v_1, \ldots, v_k\} \subset E\Gamma$ and any superset $w \supset u$, $w = \{v_1, \ldots, v_k, \ldots, v_t\}$. Show that $w \in E\Gamma$. By construction, $\psi(u) = \{\psi(v_1), \ldots, \psi(v_k)\} = \{J_{i_1}, \ldots, J_{i_k}\}$ for some m.f.s.[2] $J_{i_1}, \ldots, J_{i_k}$ of system (11.3). Then, $\psi(w) = \{J_{i_1}, \ldots, J_{i_k}, \ldots, J_{i_t}\}$. Therefore, $\bigcup_{j=1}^{k} J_{ij} = [m]$, since ψ is an isomorphism and $\psi \in E$. Therefore, $\bigcup_{j=1}^{t} J_{i_j}$ also coincides with $[m]$, which implies $\psi(w) \in E$ by Definition 11.3. Thus, $w = \psi^{-1}(\psi(w)) \in E\Gamma$.

Sufficiency: Suppose, a hypergraph $\Gamma = (V\Gamma, E\Gamma)$ for $V\Gamma = \{v_1, \ldots, v_s\}$ satisfies conditions (11.4) and (11.5). Point out an appropriate $m = m(\Gamma)$ and $D_1(\Gamma), \ldots, D_m(\Gamma)$ such that Γ is isomorphic to the m.f.s. hypergraph $G = (V, E)$ of system (11.3).

If $s = 1$, there exists a single possible option: $\Gamma = (\{v_1\}, \{\{v_1\}\})$ corresponding to the case of a feasible system. For instance, put $m = m(\Gamma) = 1$, $D_1(\Gamma) = X$ for an arbitrary $X \neq \varnothing$. Obviously, for the appropriate inclusions system, the unique m.f.s has the index set $J_1 = [m]$. Therefore, its m.f.s. hypergraph $G = (\{J_1\}, \{\{J_1\}\})$ is trivially isomorphic to Γ.

Consider the case of $s > 1$. Introduce the following notation $W = 2^{V\Gamma} \setminus (E\Gamma \cup \{\varnothing\})$. By construction, the set W is finite. Put $m = m(\Gamma) = |W|$. Further, for any $k \in [s]$, we define the set J_k as follows:

$$J_k = \{i \colon v_k \notin w_i, w_i \in W\}.$$

Verify that the following equations

$$\varnothing \neq J_k \subset [m], \tag{11.6}$$

$$J_{k_1} = J_{k_2} \iff k_1 = k_2, \tag{11.7}$$

and

$$(\forall L \subset [s], L \neq \varnothing) \left(\bigcup_{k \in L} J_k = [m] \iff \{v_k \colon k \in L\} \in E\Gamma \right). \tag{11.8}$$

Equation (11.6) follows straightforwardly from condition (11.4). Indeed, since $s > 1$, for any $k \in [s]$, there exists $k' \in [s]$, such that

[2]For the simplicity, we do not distinguish m.f.s and its index set.

$k \neq k'$. By construction of the set W, in $[m]$, there exist numbers i and i', for which $w_i = \{v_{k'}\}$ and $w_{i'} = \{v_k\}$. Since, $i \in J_k$ and $i' \notin J_k$, equation (11.6) is valid. Equation (11.7) can be proved similarly.

To prove (11.8), consider an arbitrary non-empty subset $L \subset [s]$. If $\{v_k : k \in L\} \notin E\Gamma$, then there is $i \in [m]$, such that $\{v_k : k \in L\} = w_i$. Thus, by construction of the sets J_k,

$$i \notin \bigcup_{k \in L} J_k. \tag{11.9}$$

On the other hand, let condition (11.9) be valid for some $i \in [m]$. Again, by construction of the sets J_k, we obtain that $w_i \in W$ and $\{v_k : k \in L\} \subseteq w_i$. Since $w_i \notin E\Gamma$, then, by condition (11.5), $\{v_k : k \in L\} \notin E\Gamma$ as well.

For any $j \in [m]$, define the set $D_j(\Gamma)$ as follows:

$$D_j(\Gamma) = \{k \in [s] : j \in J_k\}.$$

We show that $J_1, \ldots, J_s$ and only them are m.f.s. of system (11.3). Indeed, for any $k \in [s]$ and any $j \in J_i$, $j \in J_k$, by construction. Therefore, $\bigcap_{j \in J_k} D_j(\Gamma) \neq \varnothing$, i.e. J_k is an index set of some feasible subsystem of system (11.3). On the other hand, consider an arbitrary non-empty subset $L \subset [m]$ specifying a feasible subsystem of system (11.3). There exists $k \in [s]$, such that $k \in \bigcap_{j \in L} D_j(\Gamma)$. By construction of the sets $D_j(\Gamma)$, for any $j \in L$, $j \in J_k$, i.e. $L \subseteq J_k$. Therefore, the family of m.f.s. of system(11.3) is exhausted by the sets $J_1, \ldots, J_s$.

Finally, establish the bijection $\psi : V\Gamma \to V = \{J_1, \ldots, J_s\}$ by the trivial rule $\psi(v_k) = J_k$. By condition (11.8), ψ is a desired isomorphism of the hypergraph *Gamma* onto the hypergraph G, which is the m.f.s. hypergraph of system (11.3). Theorem is proved. □

In the sequel, any finite hypergraph without multiple edges satisfying the conditions (11.4)–(11.5) we call an m.f.s. hypergraph.

As it follows from Theorem 11.1, the class of m.f.s. hypergraphs is sufficiently large. In the sequel, for any natural q, we describe [16] a subclass of m.f.s. hypergraphs of inclusion systems having a committee generalized solution of length q.

Definition 11.4. For an m.f.s. hypergraph $\Gamma = (V\Gamma, E\Gamma)$ and natural numbers σ and τ, a finite sequence of nodes $S = (v_{i_1}, \ldots, v_{i_{\sigma+1}})$ is called (σ, τ)-simplex in the hypergraph Γ, if the inclusion $\{v_{i_j} : j \in L\} \in E\Gamma$ holds for any $L \subset [\sigma + 1]$, $|L| = \tau + 1$.

The notion of (σ, τ)-simplex stems from the geometric reasonings. Indeed, it is easy to verify that $(2, 1)$-simplex induces in the hypergraph Γ a triangle (a cycle of length 3).

Theorem 11.2. *System* (11.1) *has a committee generalized solution of length q if and only if its m.f.s. hypergraph contains a (σ, τ)-simplex for $\sigma = q - 1$ and $\tau = \lfloor (q-1)/2 \rfloor$.*

Proof. *Necessity*: Let $Q = (x^1, \ldots, x^q)$ be an arbitrary committee of system (11.1) and G be its m.f.s. hypergraph. To each entry x^i of the committee Q, assign the set $J'_i = \{j \colon x^i \in D_j\}$. Let $J_1, \ldots, J_q$ be (non-necessarily distinct) index sets of m.f.s. of system (11.1), such that $J'_i \subseteq J_i$ for any $i \in [q]$. We show that the finite sequence $(J_1, \ldots, J_q)$ is the desired (σ, τ)-simplex for $\sigma = q - 1$ and $\tau = \lfloor (q-1)/2 \rfloor$.

Indeed, consider an arbitrary subset $L \subset [q]$ of size $|L| = \tau + 1 = \lfloor (q+1)/2 \rfloor$. Without loss of generality, we assume that $L = \{1, \ldots, \tau + 1\}$. Let, for some $j \in [m]$, $j \notin \bigcup_{k=1}^{\tau} J_k$. Then, by construction, $j \notin \bigcup_{k=1}^{\tau} J'_k$. Since $\tau = \lfloor (q-1)/2 \rfloor$ and Q is a committee of length q, this leads to

$$j \in \left(\bigcap_{k=\lfloor \frac{q+1}{2} \rfloor}^{q} J'_{i_k} \right) \subseteq \left(\bigcap_{k=\lfloor \frac{q+1}{2} \rfloor}^{q} J_{i_k} \right) \subset J_{i_{(\tau+1)}} \subset \bigcup_{l=1}^{\lfloor (q+1)/2 \rfloor} J_{i_l}.$$

Therefore,

$$\bigcup_{l=1}^{\lfloor (q+1)/2 \rfloor} J_{i_l} = [m]$$

and the sequence $(J_1, \ldots, J_q)$ is (σ, τ)-simplex in the hypergraph G.

Sufficiency: Suppose there are (not necessarily distinct) m.f.s $J_1, \ldots, J_q$, such that the sequence $(J_1, \ldots, J_q)$ is a (σ, τ)-simplex in the hypergraph G for $\sigma = q - 1$ and $\tau = \lfloor (q-1)/2 \rfloor$. Consider an arbitrary $j \in [m]$. Without loss of generality, assume that $\{i \colon i \in D_j\} = \{1, \ldots, q'\}$. Then, $j \notin \bigcup_{l=q'+1}^{q} J_l$. Therefore, $q - q' < \lfloor (q+1)/2 \rfloor = \tau + 1$. Therefore, $q' \geq \lfloor q/2 \rfloor + 1$, i.e. $Q = (x^1, \ldots, x^q)$ is a committee solution of system (11.1) for any solutions x^i of the m.f.s. J_i. $\square$

11.3 Game Theoretic Conditions of Committee Existence

In this section, we propose a series of necessary existence conditions for committee generalized solutions in terms of optimal strategies for the appropriate antagonistic games between *the Nature* and *the Researcher*. Actually, for any natural numbers q, $k < q$ and any system of abstract inclusions having a generalized committee solution of length q, we propose an attainable relative lower cardinality bound for the maximum subsystem that has a committee solution of length k. In order to obtain this result, we find an upper value of the corresponding zero-sum two-player game.

For any setting of the game in question, the first player, we call him or her *Researcher*, tries to choose a finite sequence of length k attempting to enlarge the subsystem resolved by this sequence as a generalized committee solution. On the other hand, the second player, *Nature*, tries to prevent these attempts suggesting infeasible systems, which can hardly be solved with committees of length less than q.

We show that, almost always, this game has no value being set in pure strategies but has a value in mixed ones. Further, we investigate the asymptotic behavior of our bound in the case of $k = q - h$ for any fixed h and $q \to \infty$. In this section, we mainly follow the papers [20, 21].

11.3.1 *Problem statement*

Suppose we are given by some non-empty ground set X and a family of its subsets $D_1, D_2, \ldots, D_m$ defining the following abstract system of inclusions

$$x \in D_j \quad (j \in [m]), \tag{11.10}$$

and natural numbers q and $k < q$. We assume that system (11.10) has a generalized committee solution of length q and is aimed to provide a lower cardinality bound for a maximum subsystem (of system (11.10)) satisfied by a committee of length k.

To obtain the desired bound, we consider an arbitrary committee solution $Q = (x^1, \ldots, x^q)$ of system (11.10). Similar to Section 11.2, we assign to the committee Q an incidence $m \times q$-matrix $A = A(Q)$

with entries

$$a_{ji} = \begin{cases} 1, & \text{if } x^i \in D_j \\ -1, & \text{otherwise.} \end{cases}$$

Denote by a_j the jth row of the matrix A. By construction, for any a_j,

$$\sum_{i=1}^{q} a_{ji} \geq 1. \tag{11.11}$$

In the sequel, we denote by $M(q)$ the set of $m \times q$-matrices A satisfying equation (11.11).

To any subset $I \subset [q]$ of cardinality k, we assign

- the characteristic vector $\tau = \tau(I) = \sum_{e \in I} e_i^q$, where e_i^q is the ith orth of the q-dimensional Euclidean space E_q,
- the subcommittee $Q(I) = (x^i \colon i \in I)$ of length k defined by the subset I,
- maximal subsystem of system (11.10) resolved by the subcommittee $Q(ISuffi)$; without loss of generality, in the sequel, we do not distinguish this subsystem and its index set $J(I)$, which satisfies the equation $J(I) = \{j \colon (a_j, \tau(I)) \geq 1\}$, by construction,
- the relative cardinality of the subsystem $J(I)$

$$\delta_{q,k}(I, A) = \frac{|J(I)|}{m}.$$

To any matrix $A \in M(q)$, assign the number

$$\delta_{q,k}(A) = \max\{\delta_{q,k}(I, A) \colon I \subset [q], |I| = k\}. \tag{11.12}$$

Consider the following two-player zero-sum game $\Gamma = (X, Y, K)$, where strategy sets of the first and second players are

$$X = \{I \subset [q] \colon |I| = k\} \text{ and } Y = M(q)$$

respectively and the payoff function is $K(I, A) = \delta_{q.k}(I, A)$. To answer the main question of this section, we need to find an upper value

$$\delta_{q,k} = \min_{A \in M(q)} \delta_{q,k}(A) = \min_{A \in M(q)} \max_{J \subset [q], |I| = k} \delta_{q,k}(I, A)$$

of this game. For the subsequent constructions, we need the following standard notation:

$$\begin{aligned}
\binom{n}{i} &= \frac{n!}{i!(n-i)!} && \text{the binomial coefficient}\\
b(i;n,p) &= \binom{n}{i} p^i (1-p)^{n-i} && \text{the binomial distribution mass function, i.e. probability of } i \text{ successes in } n \text{ trials}\\
\varphi(x) &= \frac{1}{\sqrt{2\pi}}\, e^{-x^2/2} && \text{the standard Gaussian density}\\
\lfloor x \rfloor &\text{ and } \lceil x \rceil && \text{the floor and ceiling functions of real argument.}
\end{aligned}$$

Also, we call real-valued sequences $\{\xi_n\}$ and η_n *asymptotically equivalent* and use the notation $\xi_n \sim \eta_n$, if

$$\lim_{n\to\infty} \frac{\xi_n}{\eta_n} = 1.$$

Let $s = \lceil \frac{q+1}{2} \rceil$ and $t = \lceil \frac{k+1}{2} \rceil$. Similar to $\tau(I)$, to any subset $S \subset [q]$ of size s, we assign the characteristic vector $\sigma(S) = \sum_{i\in S} e_i^q$. Without loss of generality, we assume that the vector sets $\Sigma = \{\sigma(S)\}$ and $\Theta = \{\tau(I)\}$ are ordered lexicographically by descending and their elements σ^i and τ^j are labeled by natural numbers $i = 1, 2, \ldots, \binom{q}{s}$ and $j = 1, 2, \ldots, \binom{q}{k}$, respectively.

11.3.2 *Upper value*

In this subsection, we prove the exact formulas for computing of $\delta_{q,k}$. Further, we find a mixed strategy equilibrium for the considered game.

Theorem 11.3. *For any natural numbers $k < q$, the following equations*

$$\delta_{q,k} = \frac{s}{q} \sum_{l=t-1}^{k-1} \frac{\binom{l}{t-1}\binom{(q-1)-l}{(s-1)-(t-1)}}{\binom{q-1}{s-1}}$$

$$= \frac{k}{q} \sum_{l=t-1}^{s-1} \frac{\binom{l}{t-1}\binom{(q-1)-l}{(k-1)-(t-1)}}{\binom{q-1}{k-1}} \tag{11.13}$$

are valid.

Proof. Take an arbitrary matrix $A \in M(q)$. By construction, for any row a_j of the matrix A, there exists a number $i = i(j)$ such that the equation $(a_j, \sigma^i) = s$ holds for the vector $\sigma^i \in \Sigma$.

To any vector $\sigma^i \in \Sigma$, assign a number y_i by induction on i. Indeed, for $i = 1$, define the set $J_1 = \{j \in [m] \colon (a_j, \sigma^1) = s\}$ and set $y_1 = |J_1|/m$. Further, let J_p and y_p be specified for all $p \leq i < \binom{q}{s}$. Then, set

$$J_{i+1} = \left\{j \in [m] \colon (a_j, \sigma^{i+1}) = s\right\} \setminus \bigcup_{p=1}^{i} J_p$$

and $y_{i+1} = |J_{i+1}|/m$.

By construction, $y_i \geq 0$, $\sum_{i=1}^{\binom{q}{s}} y_i = 1$ and the following equation

$$\delta_{q,k}(I, A) = \frac{|\{j \colon (a_j, \tau(I)) \geq 1\}|}{m} \geq \sum_{i=1}^{\binom{q}{s}} \left\{y_i \colon (\sigma^i, \tau(I)) \geq t\right\} \tag{11.14}$$

is valid. Indeed, it can be seen that, if $(\sigma^i, \tau(I)) \geq t$, then, for any $j \in [m]$, the inequality $(a_j, \tau(I)) \geq 1$ follows from the condition $(a_j, \sigma^i) = s$. It should be noted that equation (11.14) is not necessarily tight. For instance, let

$$A = \left(\begin{array}{cc|ccc} 1 & 1 & 1 & 1 & -1 \\ -1 & 1 & 1 & -1 & 1 \\ 1 & 1 & -1 & 1 & -1 \end{array}\right) \in M(5),\ k = 3,\ t = 2,\ \text{and } I = \{3, 4, 5\}.$$

It is easy to verify that, in this case,

$$\sum_{i=1}^{\binom{5}{3}} \{y_i \colon (\sigma^i, \tau(I)) \geq 2\} = \sum_{i=4}^{10} y_i = \frac{1}{3}$$

(see Table 11.1), while $\delta_{5,3}(I, A) = 2/3$.

Table 11.1. Computing of y_i.

i	σ^i	J_i	y_i
1	$\{1,1,1,0,0\}$	$\{1\}$	$\frac{1}{3}$
2	$\{1,1,0,1,0\}$	$\{3\}$	$\frac{1}{3}$
3	$\{1,1,0,0,1\}$	$\emptyset$	0
4	$\{1,0,1,1,0\}$	$\emptyset$	0
5	$\{1,0,1,0,1\}$	$\emptyset$	0
6	$\{1,0,0,1,1\}$	$\emptyset$	0
7	$\{0,1,1,1,0\}$	$\emptyset$	0
8	$\{0,1,1,0,1\}$	$\{2\}$	$\frac{1}{3}$
9	$\{0,1,0,1,1\}$	$\emptyset$	0
10	$\{0,0,1,1,1\}$	$\emptyset$	0

As it follows from (11.12) and (11.14),

$$\delta_{q,k}(A) \geq \max_{I\subset[q],|I|=k} \sum_{i=1}^{\binom{q}{s}} \left\{y_i \colon (\sigma^i, \tau(I)) \geq t\right\}. \tag{11.15}$$

We proceed with finding a lower bound for the RHS of (11.15), which coincides with the optimal value β of the following linear program:

$$\begin{aligned} L: \ \beta = \min v \\ -\sum_{i\colon (\sigma^i,\tau^j)\geq t} y_i + v \geq 0 \qquad \left(j \in \left\{1,2,\ldots,\binom{q}{k}\right\}\right) \\ \sum_{i=1}^{\binom{q}{s}} y_i = 1, \quad y \geq 0. \end{aligned} \tag{11.16}$$

To find β, it is sufficient to compute $|\{i\colon (\sigma^i,\tau^j) \geq t\}|$ for any j. Without loss of generality, we consider the case $j = 1$, when

$$\tau^j = (\underbrace{1,1,\ldots,1}_{k},0,\ldots,0),$$

since the argument can be easily extended to the general case due to the obvious symmetry. We use the scheme illustrated in Figure 11.1.

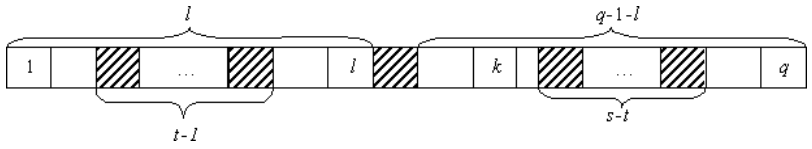


Figure 11.1. Calculation scheme for (11.17).

For any $l \in \{t-1, t, \ldots, k-1\}$, we have

$$\left|\left\{ i\colon \sum_{p=1}^{l} \sigma_p^i = t-1, \sigma_{l+1}^i = 1 \right\}\right| = \binom{l}{t-1}\binom{q-(l+1)}{s-t}. \tag{11.17}$$

Thus,

$$|\{i\colon (\sigma^i, \tau^1) \geq t\}| = \sum_{l=t-1}^{k-1} \binom{l}{t-1}\binom{q-1-l}{s-t},$$

since each σ^i, for which $(\sigma^i, \tau^1) \geq t$, is enumerated as l run from $t-1$ to $k-1$ and only once.

Further, consider the dual for (11.16)

$$L^*:\ \alpha = \max\ u$$
$$-\sum_{j\colon (\sigma^i, \tau^j) \geq t} x_j + u \leq 0 \qquad \left(j \in \left\{1, 2, \ldots, \binom{q}{s}\right\}\right)$$
$$\sum_{j=1}^{\binom{q}{k}} x_j = 1, \quad x \geq 0. \tag{11.18}$$

Applying to the calculation of $|\{j\colon (\sigma^i, \tau^j) \geq t\}|$ the similar scheme, we obtain

$$|\{j\colon (\sigma^i, \tau^j) \geq t\}| = \sum_{t-1}^{s-1} \binom{l}{t-1}\binom{q-1-l}{k-t}. \tag{11.19}$$

Indeed, assuming by analogy that $i = 1$, for any $l \in \{t-1, t, \ldots, s-1\}$, we get

$$\left|\left\{ j\colon \sum_{p=1}^{l} \tau_p^j = t-1,\ \tau_{l+1}^j = 1 \right\}\right| = \binom{l}{t-1}\binom{q-1-l}{k-t}$$

(see Figure 11.2). Then, (11.19) is obtained by summation on l.

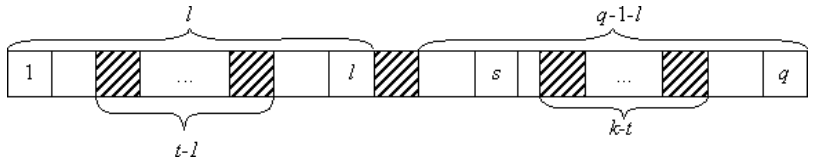


Figure 11.2. Illustration for (11.19).

Finally, consider the vectors $[\bar{x}, \bar{u}]$ and $[\bar{y}, \bar{v}]$ defined as follows:

$$\bar{x}_j = \binom{q}{k}^{-1},\ j \in \left\{1, 2, \ldots, \binom{q}{k}\right\},$$

$$\bar{y}_i = \binom{q}{s}^{-1}\ i \in \left\{1, 2, \ldots, \binom{q}{s}\right\},$$

$$\bar{u} = \binom{q}{k}^{-1} \sum_{l=t-1}^{s-1} \binom{l}{t-1}\binom{q-1-l}{k-t}$$

$$= \frac{k}{q} \sum_{l=t-1}^{s-1} \frac{\binom{l}{t-1}\binom{(q-1)-l}{(k-1)-(t-1)}}{\binom{q-1}{k-1}},$$

$$\bar{v} = \binom{q}{s}^{-1} \sum_{l=t-1}^{k-1} \binom{l}{t-1}\binom{q-1-l}{s-t}$$

$$= \frac{s}{q} \sum_{l=t-1}^{k-1} \frac{\binom{l}{t-1}\binom{(q-1)-l}{(s-1)-(t-1)}}{\binom{q-1}{s-1}}.$$

It is easy to verify that the proposed vectors $[\bar{y}, \bar{v}]$ and $[\bar{x}, \bar{u}]$ are feasible in programs L and L^* (equations (11.16) and (11.19)) respectively and satisfy the complementarity condition. Therefore, they are optimal (each in its own program) and $\bar{u} = \alpha = \beta = \bar{v}$. Thus, $\delta_{q,k}(A) \geq \bar{v} = \bar{u}$ for any $A \in M(q)$.

To prove the attainability of bound (11.13), consider the special $\binom{q}{s} \times q$ matrix $\tilde{A} = \tilde{A}(q)$, for which $a_j = 2(\sigma^j - 0.5)$. Evidently,

$\tilde{A} \in M(q)$ and, for any $I \subset [q]$ of size k, $\delta_{q,k}(I, \tilde{A}) = \beta = \bar{v}$, therefore,

$$\delta_{q,k} = \min_{A \in M(q)} \delta_{q,k}(A) \leq \delta_{q,k}(I, \tilde{A}) = \beta.$$

Theorem is proved. □

Theorem 11.3 generalizes several known results. For instance, by substitution in (11.13) $k = t = 1$, we easily obtain the following.

Corollary 11.1. *Any system* (11.10) *having a committee solution of length* q *contains a feasible subsystem of relative size at least* $\lceil \frac{q+1}{2} \rceil / q$.

On the other hand, some consequences of Theorem 11.3 seem to be surprising. For instance, for $q = 2s - 1$ and $k = 2s - 3$, we have the following lower bound

$$\delta_{2s-1,\,2s-3} = 0.5\,\frac{3s-2}{2s-1}$$

for the relative size of the maximum subsystem having a committee solution of length $q - 2$. It is evident that $\delta_{2s-1,\,2s-3}$ grows monotonically as s tends to $+\infty$ but

$$\lim_{s \to +\infty} \delta_{2s-1,\,2s-3} = 0.75 < 1. \tag{11.20}$$

It is easy to show that, for any $k < q$ (except the case $q = 2\lambda$ and $k = q - 1$), the game under the question has no value in pure strategies and

$$\max_{I \subset [q],\, |I| = k} \min_{A \in M(q)} \delta_{q,k}(I, A) = 0.$$

In Section 11.3.3, we provide a solution of this game in mixed strategies.

11.3.3 *Mixed strategy equilibrium*

To prove the existence of a mixed strategy equilibrium the game Γ, we show that its mixed extension coincides with a mixed extension of an appropriate matrix game.

Indeed, the pure strategy set of *Researcher* is finite, by construction. His (her) set $\bar{X}$ of mixed strategies is

$$\bar{X} = \left\{ x \in \mathbb{R}^{\binom{q}{k}} : \sum_{j=1}^{\binom{q}{k}} x_j = 1, x \geq 0 \right\}.$$

Investigate the pure strategy set $M(q)$ of *the Nature*. We start with exclusion from $M(q)$ the matrices that contain a row with more than $\lceil \frac{q+1}{2} \rceil$ ones, since these matrices are dominated by some other pure strategies of the second player. Further, thanks to the evident invariance of the payoff function to any permutation of rows in a matrix and simultaneous cloning of them, we proceed with the exclusion of one matrix from any couple of equivalent strategies. Without loss of generality, we assume that after such a filtering, each pure strategy of the second player is a matrix $m \times q$-matrix A whose rows are as follows:

$$\begin{aligned}
a_j &= 2(\sigma^1 - 0.5), \qquad j \in \{1, 2, \ldots, m_1\} \\
a_j &= 2(\sigma^2 - 0.5), \qquad j \in \{m_1 + 1, \ldots, m_1 + m_2\} \\
&\ldots \\
a_j &= 2(\sigma^{\binom{q}{s}} - 0.5), \qquad j \in \left\{ \sum_{i=1}^{\binom{q}{s}-1} m_i + 1, \ldots, \sum_{i=1}^{\binom{q}{s}} m_i \ (= m) \right\}
\end{aligned}$$

for some non-negative mutually co-prime[3] integers $m_1, \ldots, m_{\binom{q}{s}}$. Introducing the notation $y_i = \frac{m_i}{m}$, we assign to any matrix $A \in M(q)$ (similar to the proof of Theorem 11.3) a non-negative rational vector y normalized by the sum of its entries. It is easily to verify that such mapping is a bijection, i.e. the set of pure strategies of the second

[3]We assume that 0 is co-prime to any integer.

player coincides with the set

$$Y = \left\{ y \in \mathbb{Q}^{\binom{q}{s}} \colon \sum_{i=1}^{\binom{q}{s}} y_i = 1, y \geq 0 \right\}$$

up to an isomorphism. Therefore, we can assume that the mixed strategy set $\bar{Y}$ of this player consists of probabilistic measures defined on an arbitrary σ-algebra over Y including all its singletons $\{y\}$.

For any $y \in Y$, we denote the matrix corresponding to the vector y by $A(y)$. In particular, any orth $e_i^{\binom{q}{s}}$ of the Euclidean space $E_{\binom{q}{s}}$ corresponds to the matrix consisting of the single row σ^i. Since, for any pure strategy I_j of the first player, the payoff is

$$\delta_{q,k}(I_j, A(y)) = \sum_{i\colon\colon (\sigma^i, \tau^j) \geq t} y_i,$$

the pure strategy $A(y)$ is equivalent to the mixed strategy μ_y specified by the following equation

$$\mu_y(e_i^{\binom{q}{s}}) = y_i \qquad \left(1 \leq i \leq \binom{q}{s}\right)$$

and can be excluded. Thus, the mixed strategy set $\bar{Y}$ of the second player is isomorphic to the set Y as well.

Summarizing, we come to the conclusion that a mixed extension of the game Γ is equivalent to the mixed extension of the matrix game defined by the linear programs L and L^*. Thus, we proved the following theorem.

Theorem 11.4. *The game Γ has a mixed strategy equilibrium. The measures $\bar{x}$ and $\bar{\mu}$, where*

$$\bar{x}_j = \binom{q}{k}^{-1}, \left(1 \leq j \leq \binom{q}{k}\right)$$

and

$$\bar{\mu}(e_i^{\binom{q}{s}}) = \binom{q}{s}^{-1} \left(1 \leq i \leq \binom{q}{s}\right),$$

are optimal mixed strategies of the Researcher and the Nature, respectively, while the value $\delta_{q,k}$ is specified by equation (11.13).

It should be noted that the optimal mixed strategy of the second player does not depend on k and is determined ambiguously. For instance, the matrix $A(q)$ introduced in the proof of Theorem 11.3 is an optimal pure strategy.

11.3.4 *Asymptotic bounds*

In this section, we introduce approximate formulas to compute $\delta_{q,k}$ for large values of q assuming that $k = q - n$ for some fixed integer n.

Theorem 11.5. *Let λ be an arbitrary natural number.*

(i) *If $n = 2\lambda$, then*

$$\lim_{q\to\infty} \delta_{q,q-n} = \frac{1}{2}\,(1 + b(\lambda; 2\lambda - 1, 0.5)). \tag{11.21}$$

(ii) *If $n = 2\lambda - 1$, then the limit $\lim_{q\to\infty} \delta_{q,q-n}$ does not exist since*

$$\lim_{s\to\infty} \delta_{2s,2(s-\lambda)+1} = \frac{1}{2} + b(\lambda; 2\lambda - 1, 0.5) \quad \text{and} \tag{11.22}$$

$$\lim_{s\to\infty} \delta_{2s-1,2(s-\lambda)} = \frac{1}{2}. \tag{11.23}$$

For the sake of brevity, we skip the proof, which can be found in [21].

Corollary 11.2. *For any fixed natural k and $q > k$, the following equations*

$$\delta_{q,k} \geq 1/2 \quad \text{and} \quad \lim_{q\to\infty} \delta_{q,k} = \frac{1}{2}$$

are valid.

Corollary 11.3. *Limit equations* (11.21) *and* (11.22) *depend asymptotically on λ as follows:*

$$\lim_{q\to\infty} \delta_{q,q-2\lambda} \sim \frac{1}{2} + \frac{1}{\sqrt{2\lambda - 1}}\,\varphi\left(\frac{1}{\sqrt{2\lambda - 1}}\right)$$

and

$$\lim_{s\to\infty} \delta_{2s,2(s-\lambda)+1} \sim \frac{1}{2} + \frac{2}{\sqrt{2\lambda - 1}}\,\varphi\left(\frac{1}{\sqrt{2\lambda - 1}}\right).$$

Table 11.2. Numeric evaluation, part 1.

λ	1	2	3	4	5
$\lim\limits_{q\to\infty} \delta_{q,q-2\lambda}$	0.750	0.688	0.656	0.637	0.623
Gaussian appr.	0.742	0.695	0.661	0.640	0.626
relative error	0.011	0.010	0.008	0.005	0.005
λ	6	7	8	9	10
$\lim\limits_{q\to\infty} \delta_{q,q-2\lambda}$	0.613	0.605	0.598	0.593	0.588
Gaussian appr.	0.615	0.606	0.600	0.594	0.589
relative error	0.003	0.003	0.003	0.002	0.002

Table 11.3. Numeric evaluation, part 2.

λ	1	2	3	4	5
$\lim\limits_{s\to\infty} \delta_{2s,2(s-\lambda)+1}$	1.000	0.875	0.812	0.773	0.746
Gaussian appr.	0.984	0.891	0.823	0.781	0.752
relative error	0.016	0.018	0.014	0.010	0.008
λ	6	7	8	9	10
$\lim\limits_{s\to\infty} \delta_{2s,2(s-\lambda)+1}$	0.726	0.709	0.696	0.685	0.676
Gaussian appr.	0.730	0.713	0.699	0.688	0.678
relative error	0.006	0.006	0.004	0.004	0.003

Proof. The provided asymptotic equations follow straightforwardly from the celebrated de Moivre–Laplace theorem. Indeed, according to this theorem, the equation $b(i;n,p) \sim \frac{1}{\sqrt{np(1-p)}} \varphi\left(\frac{i-np}{\sqrt{np(1-p)}}\right)$ holds uniformly by i. In our case, $n = 2\lambda - 1$, $i = \lambda$, $p = \frac{1}{2}$, which completes the proof. □

It is notable that the Gaussian approximation for the biggest value of the binomial distribution probability mass function performs well even for rather small values of the parameter λ. In particular, this is confirmed by the numerical data presented in Table 11.2 and Table 11.3.

11.3.5 *Concluding remarks*

Remark 11.1. The result of Theorem 11.3 is valid for an arbitrary system of inclusions (11.10) and can be possibly improved for more specific cases, e.g. for systems of linear inequalities. This issue still remains open.

Remark 11.2. Since bounds (11.13) have a flavor of cumulative probability function for the hypergeometric distribution, Theorem 11.5 can be considered beyond the context of committee solutions, as a limit theorem.

Remark 11.3. Results of this section can have curious practical applications. For instance, according to the common sense, we could expect that $\lim_{q\to\infty} \delta_{q,q-2} = 1$, but it is not so (see (11.20)). This fact can be interpreted in terms of voting theory. Indeed, suppose a parliament of some fictitious country makes decisions by a simple majority voting. In some working day, the parliament should consider some package of bills. Suppose, we know that, if all the deputies will vote, each bill in the package will be accepted. As a consequence of the mentioned result, we come to conclusion that, in the worst case, if at least two deputies will abstain from voting while the others keep their votes, the parliament can accept at most 75% of the initial package.

11.4 Affine Separating Committees and Ensembles of Linear Classifiers

In this section, we investigate properties of committee solutions of infeasible system of linear inequalities, which is a special kind of abstract system of constraints (11.1). In this case, committee solutions are closely related to the special type of learning algorithms known in the literature as *ensemble learning techniques*.

We start with the common setting of the two-pattern classification problem (see, e.g. [4]). Suppose, we are given by a probabilistic triple $(X \times Y, \mathcal{A}, P)$. To this end, X is a feature space and $Y = \{-1, 1\}$ is the set of class labels. In many cases, we can assume that X is a subset of the n-dimensional Euclidean space E_n. It is required, in the preliminary given family of classifiers $\mathcal{H} \subset [X \to Y]$, to find "the

most accurate" $\bar{h}$. An accuracy criterion admits numerous formalizations. For the sake of simplicity, we focus on the following one $\bar{h} = \arg\min\{P(f(x) \neq y)\colon h \in \mathcal{H}\}$, i.e. on the finding of a classifier that minimizes the probability of misclassification.

If the probabilistic measure is known, this problem has the well-known closed form solution — the Bayes classifier. In the general case considered in this section, when all information about the unknown measure P is exhausted by the finite i.i.d. *training* sample

$$(x_1, y_1), \ldots, (x_m, y_m), \tag{11.24}$$

the goal is to propose an efficient *learning* algorithm that could find a good approximation to the desired optimal classifier. In the context of the famous Vapnik–Chervonenkis structural risk minimization learning approach, it is significant to design learning algorithms minimizing frequency of misclassification on sample (11.24) regularized by a capacity of the family $\mathcal{H}$ in terms of its VC-dimension.

We consider the setting of such a learning problem, where it is required to fit a piecewise linear classifier

$$h(x) = \operatorname{sign} \sum_{j=1}^{k} \alpha_j \operatorname{sign}(c_j^T x - d_j) \tag{11.25}$$

for some non-negative weights α_j, which without loss of generality can be assumed integers, vectors c_j and real biases d_j. In the literature (see, e.g. [32]) classifier (11.25) is called an *affine separating committee*. The motivation of studying such a classifiers stems from the following points:

(i) For any non-contradictory[4] sample (11.24), there exists a *perfect* affine committee classifier making no classification errors on this sample [38].
(ii) The family of affine committees (11.25) defined over the n-dimensional feature space E_n and sharing the property $\sum_{j=1}^{k} \alpha_j = q$ has bounded VC-dimension [33].

[4]For which the condition $x_{i_1} = x_{i_2}$ implies $y_{i_1} = y_{i_2}$.

We proceed with the following notation. Let subsets A and B be defined (by sample (11.24)) as follows:

$$A = \{x_i \colon y_i = 1\}, \quad B = \{x_i \colon y_i = -1\}. \tag{11.26}$$

Any classifier h determined by equation (11.25) can be equivalently represented by the following finite sequence $K = K(h) = (f_1, \ldots, f_q)$, such that $q = \sum_{j=1}^{k} \alpha_j$ and

$$\begin{aligned} f_1(x) \equiv \cdots \equiv f_{\alpha_1}(x) &\equiv c_1^T x - d_1, \\ f_{\alpha_1+1}(x) \equiv \cdots \equiv f_{\alpha_1+\alpha_2}(x) &\equiv c_2^T x - d_2, \\ &\cdots \\ f_{q-\alpha_k+1}(x) \equiv \cdots \equiv f_q(x) &\equiv c_k^T x - d_k. \end{aligned}$$

It is easy to see that an affine separating committee is a natural generalization of the concept of a separating hyperplane in Euclidean spaces. Indeed, by means of the famous hyperplane separation theorem (see, e.g. [5]), for any finite sets A and B, the equation $\mathrm{conv}(A) \cap \mathrm{conv}(B) = \varnothing$ implies the existence of a linear function $f(x) = c^T x - d$ such that the hyperplane $H = \{x \in E_n \colon c^T x - d = 0\}$ separates these sets, i.e. $f(a) > 0$ and $f(b) < 0$ for any $a \in A$ and $b \in B$, respectively. Thus, if the sets A and B are separable in the regular case, then there exists an affine committee of length 1 that separates them. For the general case, the following criterion is valid.

Theorem 11.6 ([38]). *Finite subsets $A, B \subset E_n$ can be separated by an affine committee if an only if $A \cap B = \varnothing$.*

Actually, the sufficiency part of the proof of Theorem 11.6 is rather constructive and provides an algorithm for construction of an affine separating committee for the sets A and B of length at most $|A \cup B|$.

To diminish the possible "overfitting" of committee classifiers, it is important, for any training sample (11.24) and the corresponding sets (11.26), to find affine separating committee of the least possible length. Such committee is called *minimum*. Besides that, the problem of finding the minimum committees is of interest in computational geometry. Indeed, since, for given set A and B, any committee of length q induces affine separating committee of length $q + 2t$ for the

same sets and any natural t, the length of a minimum committee is a quantitative characteristic of the separability (or *inseparability*) of these sets.

Problem 11.1 (Minimum Affine Separating Committee (MASC) For give sets $A, B \subset E_n$, it is required to find an affine separating committee of the minimum length

In the conclusion of this chapter, we present a short overview of the recent results concerning algorithmic analysis of the MASC problem following [32].

Theorem 11.7 ([25]). *The minimum affine separating committee problem is strongly NP-hard and remains intractable even in the case, when* $A \cup B \subset \{x \in \{0,1,2\}^n : \|x\|_2 \leq 2\}$. *The MASC problem does not belong to the APX approximability class, unless* $P \neq NP$.

As it follows from Theorem 11.7, the MASC problem is hard to solve not only in the class of exact algorithms but even with any constant approximation ratio. The following theorem extends this result to the spaces of any fixed dimension.

Theorem 11.8 ([33]). *The MASC problem is polynomially solvable in the real line and strongly NP-hard in* n*-dimensional Euclidean space for any fixed dimension* $n > 1$.

It should be noted that the claim of Theorem 11.8 remains valid even in the case when the set $A \cup B$ is in the *general position*. As usual, a finite set $D \subset E_n$ of size $|D| > n$ is said to be *in general position*, if, for any $D' \subset D$, $|D'| = n+1$, the dimension of the affine hull $\mathrm{aff}(D')$ is equal to n. The special setting of the MASC problem given in the n-dimensional Euclidean space with additional condition on general position of $A \cup B$ is called MASC-GP(n).

Almost all known results in the field of efficient algorithm construction for the MASC problem are based on the following theorem, which can be regarded as a specification of Theorem 11.1 to the case, when training sets are in general position.

Theorem 11.9 ([30]). *For any finite subsets* $A, B \subset E_n$ *being in general position, for which* $A \cap B = \varnothing$ *and* $|A \cup B| = m$, *there exists*

an affine separation committee of length

$$q \leq 2 \left\lceil \frac{\lfloor (m-n) \rfloor}{n} \right\rceil + 1. \tag{11.27}$$

The proof of Theorem 11.9 mainly follows from the two subsequent geometric properties of finite-dimensional Euclidean spaces.

Property 11.1. Let Z be a finite subset of E_n and $\varnothing \neq Z' \subset Z$ such that $|Z'| \leq n$ and be in general position. Then, there exist open half-spaces $L_1 = \{x \colon c_1^T - d_1 < 0\}$ and $L_2 = \{x \colon c_2^T - d_2 < 0\}$ such that $Z \subset L_1 \cup L_2$ and $Z' \subset L_1 \cap L_2$.

Property 11.2. Let A and B be non-empty finite subsets of E_n, where $A \cup B$ is in general position and of size $m > n$. Then, for any subsets $A' \subset A$ and $B' \subset B$ of common size $A' \cup B' = n$ there exist $A'' \supseteq A'$ and $B'' \supseteq B'$ and a function $f(x) = c^T x - d$, such that $f(a) > 0$ and $f(b) < 0$ for any $a \in A''$ and $b \in B''$, respectively, and $|A'' \cup B''| \geq \lceil (m+n)/2 \rceil$.

Actually, the proof of Theorem 11.9 extends the proof of Theorem 11.6 and proposes the following Algorithm 1. As can be shown, Algorithm 1 is a polynomial time approximation algorithm for the problem MASC-GP(n) with time complexity bound $O(m/n \times T_n)$ and the approximation ratio $O(m/n)$. Here, T_n denotes the complexity of solving a Kramer system of linear equations over n variables.

Remark 11.4. Bound (11.27) is tight. In particular, it is attained on sets introduced in [15] and known as *uniformly distributed sets*. In [24], it is shown that the MASC problem is polynomially solvable over such sets. The formal definition is as follows.

Definition 11.5. A finite set $Z = A \cup B \subset E_n$ is called uniformly distributed (by Gale) if $A \cap B = \varnothing$, $|A \cup B| = n + 2k$ for some natural k and, for any non-trivial hyperplane $H = \{x \in E_n \colon f(x) \equiv c^T x - d = 0\}$, there exist $A' \subset A$ and $B' \subset B$, $|A' \cup B'| \geq k$, such that $f(a) > 0$ and $f(b) < 0$ for any $a \in A'$ and $b \in B'$, respectively.

It is known that, for any natural numbers n and k, there exist a uniformly distributed subset $Z = A \cup B \subset E_n$ of size $2k + n$. Therefore, in terms of machine learning, we can say that, any time, when

Algorithm 1 Affine separating committee construction algorithm

Input: a finite set $Z = A \cup B \in E_n$ being in general position.

Output: an affine separating committee for the given sets A and B.

1: for an arbitrary n-subset $A' \cup B'$ of the given set $A \cup B$, find an affine function f_0 according to Property 11.2
2: assign $Z = (A \setminus A') \cup (B \setminus B')$ and $k = 0$;
3: **while** $Z \neq \varnothing$ **do**
4: assign $k = k + 2$
5: for some (at most) n-subset $Z'' = A'' \cup B''$, $A'' \subset A$, $B'' \subset B$, take an arbitrary pair of affine functions $f_{k-1} = c_{k-1}^T x - d_{k-1}$ and $f_k = c_k^T x - d_k$ satisfying the system

$$\begin{cases} (f_{k-1}(a) > 0) \wedge (f_k(a) > 0), (a \in A''), \\ (f_{k-1}(b) < 0) \wedge (f_k(b) < 0), (b \in B''), \\ (f_{k-1}(a) > 0) \vee (f_k(a) > 0), (a \in A \setminus A''), \\ (f_{k-1}(b) < 0) \vee (f_k(b) < 0), (b \in B \setminus B''); \end{cases}$$

(the existence of such a pair is guaranteed by Property 11.1; for technical details, see [30]);
6: assign $Z = Z \setminus Z''$;
7: **end while**
8: output the affine separating committee defined by the sequence $(f_0, f_1, \ldots, f_k)$.

a training sample is defined by a uniformly distributed subset, Algorithm 1, in the family of the smallest VC dimension, in linear time with respect to the sample length, will find a committee classifier (11.25) making no classification errors.

To date, the state-of-the-art approximation algorithms for the MASC problem are based on the synthesis of the above-described approach and the celebrated multiple weights update technique (see, e.g. [3]). Characteristics of the *BoostedGreedyCommittee* algorithm [32] having the best known approximation factor are presented in the following theorem.

Theorem 11.10. *BoostedGreedyCommittee finds an* $O(((m \ln m)/n)^{1/2})$*-approximate solution for the MASC problem in time* $m^{O(n)}$.

If, for the given sets A and B, there is a minimum committee $(f_0, f_1, \ldots, f_{q-1})$ such that, for any $t = 1, \ldots, (q-1)/2$ and any $a \in A$, $b \in B$, the following equation $(f_{2t-1}(a) > 0 \vee f_{2t}(a) > 0) \wedge (f_{2t-1}(b) < 0 \vee f_{2t}(b) < 0)$ is valid, then the approximation factor of this algorithm is $O(\ln m)$.

11.5 Conclusion

Committee solutions were introduced in the middle of the 1960s as a concept generalizing the notion of regular solutions for inconsistent systems of linear inequalities evolving to fast developing field of mathematical programming, machine learning, computational geometry, and operations research.

This survey does not pretend to be called exhaustive. We intentionally restrict ourselves to some theoretic results concerning the committees, leaving without considering their applications to numerous practical decision-making problem in industry, economy, and medicine. The interested reader can examine the recent results obtained in this area, for instance, in the following publications [2, 13, 37, 46].

References

[1] Ablow, D. and Kaylor, C. (1965) A committee solution of the pattern recognition problem (corresp.) *IEEE Transactions on Information Theory*, 11, 453–455.

[2] Akberdina, V., Chernavin, N., and Chernavin, F. (2018) Application of the committee machine method to forecast the movement of exchange rates and oil prices. *Digest Finance*, 23(1), 108–120.

[3] Arora, S., Hazan, E., and Kale, S. (2012) The multiplicative weights update method: A meta-algorithm and applications. *Theory of Computing*, 8(1), 121–164.

[4] Bishop, C.M. (2007) *Pattern Recognition and Machine Learning*. Information Science and Statistics (Springer, Singapore).

[5] Boyd, S. and Vandenberghe, L. (2009) *Convex Optimization* (Cambridge University Press, Cambridge, UK).

[6] Breiman, L. (1996) Bagging predictors. *Machine Learning*, 24(2), 123–140.

[7] Breiman, L. (2001) Using iterated bagging to debias regressions. *Machine Learning*, 45, 261–277.

[8] Cortes, C. and Vapnik, V. (1995) Support-vector networks. *Machine Learning*, 20, 273–297.
[9] Dyukova, E.V., Zhuravlev, Y.I., and Rudakov, K.V. (1997) On the algebrological synthesis of correct recognition procedures based on elementary algorithms. *Computational Mathematics and Mathematical Physics*, 36(8), 1161–1167.
[10] Eremin, I. (2002) Theory of Linear Optimization. *Inverse and ill-posed problems series*, Vol. 29 (VSP, Netherlands).
[11] Eremin, I., Mazurov, V., and Astafiev, N. (1983) *Improper Linear and Convex Programming Problems* (Nauka, Moscow).
[12] Freund, Y. (1995) Boosting a weak learning algorithm by majority. *Information and Computation*, 121, 256–285. DOI: 10.1006/inco.1995.1136.
[13] Gainanov, D. and Berenov, D. (2017) Algorithm for predicting the quality of the product of metallurgical production. *CEUR Workshop Proceedings*, Aachen, Vol. 1987, pp. 194–200. http://ceur-ws.org/Vol-1987/paper29.pdf.
[14] Gainanov, D.N., Novokshenov, V.Y., and Tyagunov, L.I. (1983) Graphs generated by inconsistent systems of linear inequalities. *Mathematical Notes of the Academy of Sciences of the USSR*, 33(2), 146–150.
[15] Gale, D. (1956) Neighboring vertices on a convex polyhedron. *Linear Inequalities and Related System*, Vol. 38, pp. 255–263.
[16] Hachai, M. (1997) Classification of committee solutions of majority. *Pattern Recognition and Image Analysis*, 7(2), 260–265.
[17] Hofmann, T., Schölkopf, B., and Smola, A. (2008) Kernel methods in machine learning. *The Annals of Statistics*, 36(3), 1171–1220.
[18] Khachai, M. (1997) Estimate of the number of members in the minimal committee of a system of linear inequalities. *Computational Mathematics and Mathematical Physics*, 37(11), 1356–1361.
[19] Khachai, M. (1997) On the existence of majority committee. *Discrete Mathematics and Applications*, 7(4), 383–397.
[20] Khachai, M. (2001) A relation connected with a decision making procedure based on majority vote. *Doklady Mathematics*, 64(3), 456–459.
[21] Khachai, M. (2002) A game against nature related to majority vote decision making. *Computational Mathematics and Mathematical Physics*, 42(10), 1547–1555.
[22] Khachai, M. (2003) On approximate algorithm for minimal committee of a system of linear inequalities. *Pattern Recognition and Image Analysis*, 13(3), 459–464.
[23] Khachai, M. (2006) Computational complexity of the minimal committee problem and adjacent problems. *Pattern Recognition and Image Analysis*, 16(4), 700–710.
[24] Khachai, M. (2006) Computational complexity of the minimum committee problem and related problems. *Doklady Mathematics*, 73, 138–141.
[25] Khachai, M. (2008) Computational and approximational complexity of combinatorial problems related to the committee polyhedral separability of finite sets. *Pattern Recognition and Image Analysis*, 18(2), 236–242.

[26] Khachai, M. (2010) Computational complexity of recognition learning procedures in the class of piecewise-linear committee decision rules. *Automation and Remote Control*, 72(3), 528–539.

[27] Khachai, M. (2011) Computational complexity of combinatorial optimization problems induced by collective procedures in machine learning. *Proceedings of the Steklov Institute of Mathematics*, 272(Suppl. 1), S46–S54.

[28] Khachai, M. and Poberii, M. (2012) Computational complexity of combinatorial problems related to piecewise linear committee pattern-recognition learning procedures. *Pattern Recognition and Image Analysis*, **22**(2), 278–290.

[29] Khachai, M. and Poberii, M. (2014) Scheme of boosting in the problems of combinatorial optimization induced by the collective training algorithms. *Automation and Remote Control*, 75(4), 657–667.

[30] Khachai, M. and Rybin, A. (1998) A new estimate of the number of members in a minimum committee of a system of linear inequalities. *Pattern Recognition and Image Analysis*, 8, 491–496.

[31] Khachay, M. (2007) On the computational complexity of the minimum committee problem. *Journal of Mathematical Modelling and Algorithms*, 6(4), 547–561.

[32] Khachay, M. (2015) Committee polyhedral separability: Complexity and polynomial approximation. *Machine Learning*, 101(1–3), 231–251.

[33] Khachay, M. and Poberii, M. (2009) Complexity and approximability of committee polyhedral separability of sets in general position. *Informatica*, 20(2), 217–234.

[34] Khachay, M., Pobery, M., and Khachay, D. (2015) Integer partition problem: Theoretical approach to improving accuracy of classifier ensembles. *International Journal of Artificial Intelligence*, 13(1), 135–146.

[35] Kobylkin, K. (2012) Constraint elimination method for the committee problem. *Automation and Remote Control*, 73, 355–368.

[36] Kobylkin, K.S. (2015) Lower bounds for the number of hyperplanes separating two finite sets of points. *Proceedings of the Steklov Institute of Mathematics*, 289(1), 126–138.

[37] Lebedeva, E., Kobzeva, N., Gilev, D., Kislyak, N., and Olesen, J. (2017) Psychosocial factors associated with migraine and tension-type headache in medical students. *Cephalalgia*, 37(13), 1264–1271.

[38] Mazurov, V. (1971) Committees of inequalities systems and the pattern recognition problem. *Kibernetika*, 3, 140–146.

[39] Mazurov, V. (1990) *Committee Method in Problems of Optimization and Classification* (Nauka, Moscow).

[40] Mazurov, V. and Khachai, M. (2004) Committees of systems of linear inequalities. *Automation and Remote Control*, 65, 193–203.

[41] Mazurov, V. and Khachai, M. (2007) Parallel computations and committee constructions. *Automation and Remote Control*, 68, 912–921.

[42] Mazurov, V., Khachai, M., and Rybin, A. (2002) Committee constructions for solving problems of selection, diagnostics and prediction. *Proceedings of the Steklov Institute of Mathematics*, (Suppl. 1), S67–S101.

[43] Mazurov, V., Khachay, M., and Poberii, M. (2008) Combinatorial optimization problems related to the committee polyhedral separability of finite sets. *Proceedings of the Steklov Institute of Mathematics*, 263(2), S93–S107.
[44] Nilsson, N. (1965) *Learning Machines: Foundations of Trainable Pattern Classifying Systems* (McGraw-Hill, New York).
[45] Osborne, M. (1977) Seniority logic: A logic for a committee machine. *IEEE Transactions on Computers*, 26(12), 1302–1306.
[46] Pandey, T.N., Jagadev, A.K., Dehuri, S., and Cho, S.-B. (2018) A novel committee machine and reviews of neural network and statistical models for currency exchange rate prediction: An experimental analysis. *Journal of King Saud University — Computer and Information Sciences*. https://doi.org/10.1016/j.jksuci.2018.02.010.
[47] Rudakov, K. and Trofimov, S. (1988) An algorithm for synthesizing correct recognition procedures for problems with non-intersecting classes. *USSR Computational Mathematics and Mathematical Physics*, 28(5), 102–104.
[48] Rybin, A. (2000) On some sufficient conditions of existence of a majority committee. *Pattern Recognition and Image Analysis*, 10(3), 297–302.
[49] Schapire, R. (1990) The strength of weak learnability. *Machine Learning*, 5(2), 197–227.
[50] Schapire, R. and Freund, Y. (2012) *Boosting: Foundations and Algorithms* (MIT Press, Cambridge, Massachusetts London, England).
[51] Schölkopf, B. and Smola, A. (2002) *Learning with Kernels: Support Vector Machines, Regularization, Optimization and Beyond* (MIT Press, Cambridge, Massachusetts London, England).
[52] Tyagunov, L. (1973) On revealing the sequence of maximal feasible subsystems of an infeasible system of linear inequalities (VINITI, no. 7467-73), pp. 152–162.
[53] Vorontsov, K. (2000) Optimization methods for linear and monotone correction in an algebraic approach to the recognition problem. *Computational Mathematics and Mathematical Physics*, 40(1), 159–168.
[54] Zhuravlev, Y.I. (1977a) Correct algebras over sets of incorrect (heuristic) algorithms. I. *Cybernetics*, 13(4), 489–497.
[55] Zhuravlev, Y.I. (1977) Correct algebras over sets of incorrect (heuristic) algorithms. II. *Cybernetics*, 13(6), 814–821.
[56] Zhuravlev, Y.I. (1978) Correct algebras over sets of incorrect (heuristic) algorithms. III. *Cybernetics*, 14(2), 188–197.

© 2025 World Scientific Publishing Company
https://doi.org/10.1142/9789811267215_0012

Chapter 12

Some Collective Methods in Cluster Analysis and Regressions

Vladimir V. Ryazanov

Federal Research Center "Computer Science and Control" of the Russian Academy of Sciences, Moscow, the Russian Federation

Some methods for constructing solutions of cluster analysis problems based on processing a set of solutions for the cluster analysis problem by a set of heuristic clustering algorithms, as well as regression analysis problems using a set of different supervised classification algorithms, are considered. The solution to the problem of cluster analysis is defined as the construction of a set of equivalent information matrices. Two information matrices are called equivalent to each other if they are equaled up to a permutation of the columns. The construction of the optimal committee solution of the problem of cluster analysis by a team of algorithms is reduced to solving one discrete optimization problem on permutations. An upper bound for the variation of the basic functional is obtained. An effective (polynomial) algorithm is proposed for its minimization. Heuristic methods for constructing collective clustering are considered.

The problem of restoring a function (regression recovery) as an application of some function (corrector) to the set of solutions of special supervised classification problems is considered. A preset training sample of feature descriptions of objects is assumed. In this case, all the problems associated with the heterogeneity of the features, their different informative or incomplete sampling, are failed on the classification problems. Based on this sample and the values of the dependent variable, a random collective of classification tasks is constructed. For a recognizable object,

the class is computed, to which it belongs according to each algorithm. Then the corrector found earlier is applied to the class numbers. Two types of correctors are considered: the Bayesian and the average linear corrector. In the Bayesian corrector, the naive Bayesian method is used. The average linear corrector consists of calculating a linear combination of the mean values of the dependent value from the calculated values of the classes.

Keywords: Image analysis, cluster analysis, Bayesian method, linear corrector, regression analysis, regression recovery.

12.1 Introduction

We consider the following well-known statement. Considered a lot of objects $M = \{\mathbf{x}\}$. We assume that there is the number n of some basic properties (features) of an object $\mathbf{x}$. For simplicity, $\mathbf{x} \in R^n$. Each object has a main property y, and it is assumed that it can be calculated by $\mathbf{x}$: $y = f(\mathbf{x})$. Let the values of this dependence be known for m objects. That is, a training sample is given. Required to calculate y for each $\mathbf{x}$. In the case $y \in \{1, 2, \ldots, l\}$, we are dealing with the task of supervised classification or recognition. When the training set is unlabeled (no information about y), and perhaps l not known, the clustering problem (unsupervised classification) arises. In the case of $y \in (a, b)$, where (a, b) is a restricted set, a regression problem arises. These statements can be considered as interpolation problems. We assume further that usually the training samples are self-consistent. How to improve the quality of solving classification problems (supervised or unsupervised) and recovery of regressions, having several different models or algorithms available? In this chapter, some methods for constructing solutions to cluster analysis problems, based on processing a set of solutions for the cluster analysis problem by a set of heuristic clustering algorithms, as well as regression analysis problems using a set of different supervised classification algorithms are considered.

Consider the task of supervised classification.

Let the training sample $y_i, \mathbf{x}_i, i = 1, 2, \ldots, m, y \in \{1, 2, \ldots, l\}$ be given. According to the training sample, a supervised classification algorithm A is constructed, which classifies an arbitrary one $\mathbf{x}$ to the same class $K_i, i = 1, 2, \ldots, l$ or refuses to classify it. In the supervised classification, the heuristic degree of belonging $\mathbf{x}$ to each

of the classes (grade for class, *a posteriori* probability, etc.) is calculated, the quality of the algorithm is estimated, for example, as the probability of correct classification. If we have a set of Nsupervised classification algorithms $A_1, A_2, \ldots, A_N$, then it is possible to build a collective algorithm and a collective solution for $\mathbf{x}$. These methods have been proposed long ago [8,9,17,18]. Currently, there are monographs (for example, [1, 7]) and conferences on combining pattern classifiers. Nevertheless, what must we do if there are N solutions to the cluster analysis problem (or unsupervised classifications) for the same number of clusters for the same sample of objects? After all, the quality of each clustering is evaluated generally by different indices [3].

Consider a simple example. Let the clustering problem for a sample of $m$objects be solved into l clusters. Renumber them in any order and assign to the objects in them the labels of the corresponding clusters: $\mathbf{x}_i \xrightarrow{A_\nu} K_\alpha, i = 1, 2, \ldots, m, \nu = 1, 2, \ldots, N, \alpha = j_1, j_2, \ldots, j_l.\boldsymbol{\alpha} = \langle j_1, j_2, \ldots, j_l \rangle$ is a permutation of $\langle 1, 2, \ldots, l \rangle$. Similarly, we write the other solutions. Let $N < l!$, and other clusters $K_\alpha^i, K_\beta^j, i, j = 1, 2, \ldots, N, i \neq j, \beta \neq \alpha$ be obtained. Let clusters K_α^i and K_β^j of different solutions $i \neq j, \beta \neq \alpha$ consist of the same objects, i.e. clusters are equal. Then for each solution No $i, i = 1, 2, \ldots, N$ there is a permutation of numbers $\langle 1, 2, \ldots, l \rangle$, at which the clustering numbers will coincide with the clustering numbers of the first solution. This principle of coincidence (or "proximity") of various objects will form the basis for the construction of a collective solution.

In this chapter, a solution to the problem of cluster analysis is defined as the construction of a set of equivalent information matrices. Two information matrices are called equivalent to each other if they are equaled up to a permutation of the columns. The construction of the optimal committee solution of the problem of cluster analysis by a team of algorithms is reduced to solving one discrete optimization problem on permutations. An upper bound for the variation of the basic functional is obtained. An effective (polynomial) algorithm is proposed for its minimization. Heuristic methods for constructing collective clusterings and a video method for solving them are considered [2, 10–12].

Another approach is proposed to construct regressions using a team of methods for solving the supervised classification problem. The problem of restoring a function (regression recovery) as an

application of some function (corrector) to the set of solutions of special supervised classification problems is considered. A preset training sample of feature descriptions of objects is assumed. In this case, all the problems associated with the heterogeneity of the features, their different informative or incomplete sampling, are filled on the classification problems. Based on this sample and the values of the dependent variable, a random collective of supervised classification tasks is constructed. Here, some partial precedence algorithms [19] have been used, namely, voting algorithms on the logical regularities of classes [13, 20]. For a recognizable object, the class is computed, to which it belongs according to each algorithm. Then the corrector found earlier is applied to the class numbers. Two types of correctors are considered: the Bayesian and the average linear corrector. In the Bayesian corrector, the naive Bayesian method is used. The average linear corrector consists of calculating a linear combination of the mean values of the dependent value from the calculated values of the classes. The idea of correctness in the training data was used as a justification. It was shown that constructed under certain conditions for the number of classes of supervised classification algorithms, the number of algorithms, and the length of a training consistent sample, there is an error-free recovery of dependencies $y = f(\mathbf{x})$ on the training data [14–16].

12.2 Collective Methods for Solving the Cluster Analysis Problem

Let the training set $\mathrm{X} = \{\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_m\}$ be given.

Definition 12.1. We call equivalent matrices $I = \|\alpha_{ij}\|_{m\times l}, \alpha_{ij} \in \{0, 1\}$ and $I' = \|\alpha'_{ij}\|_{m\times l}, \alpha'_{ij} \in \{0, 1\}$ if they are equal up to permutation of columns.

This definition defines the class of equivalent matrices for some matrix.

Definition 12.2. A clustering algorithm is an algorithm $A^c(\mathrm{X}) = \mathrm{K}(\|\alpha_{ij}\|_{m\times l})$ that associates a set of equivalent information matrices with a sample X.

Hereinafter, Section 12.2 uses the definitions of [11].

Here, $\alpha_{ij} = 1$ means that the object $\mathbf{x}_i$ is related to the cluster K_j by some clustering algorithm. The number of clusters and the length of the training sample are considered given. This definition emphasizes the fact that in an arbitrary partitioning of the sample into l clusters, we have complete freedom in the numbering of the clusters. In the following, we always consider dimension matrices $m \times l$.

Let N clustering algorithms $A_1^c, A_2^c, \ldots, A_N^c$ and results of their application to the sample X be given. $A_\nu^c(\mathrm{X}) \in \mathrm{K}(\|\alpha_{ij}^v\|_{m\times l})$ are clusters of the training sample. Denote $I_\nu = \|\alpha_{ij}^\nu\|_{m\times l}$.

So, we have $\mathrm{I} = \mathrm{K}(I_1) \times \mathrm{K}(I_2) \times \cdots \times \mathrm{K}(I_n)$ or set $\mathrm{I} = \{(I_1', I_2', \ldots, I_n'), I_\nu' \in \mathrm{K}(I_\nu)\}$.

There are two tasks:

1. Building a mapping I on $\mathrm{K}_c, \mathrm{I} \to \mathrm{K}_c = \{\mathrm{K}\|c_{ij}\|_{m\times l}\}, c_{ij} \in \{0, 1\}$, i.e. building some kind of clustering.
2. Finding the optimal element in K_c, i.e. finding the best clustering in K_c.

Definition 12.3. An operator $\mathrm{B}(I_1', I_2', \ldots, I_N') = B = \|b_{ij}\|_{m\times l}$ is called an adder if $b_{ij} = \sum_{\nu=1}^N \alpha_{ij}^{\prime\nu}$.

It's clear that $0 \le b_{ij} \le N, b_{ij} \in \{0, 1, 2, \ldots, N\}$.

Definition 12.4. The operator r is called the threshold decision rule, if $r(B) = C = \|c_{ij}\|_{m\times l}, c_{ij} = \begin{cases} 1, & b_{ij} \ge \delta_i, \\ 0, & \text{otherwise.} \end{cases}$

Here $\delta_i \in R^n.\delta_i, i = 1, 2, \ldots, m$ are the parameters of the collective algorithm. For example, $\delta_i = \theta \sum_{j=1}^l b_{ij}, 0 < \theta < 1$.

Definition 12.5. The committee-based synthesis of the information matrix C on an element $\tilde{I}' = (I_1', I_2', \ldots, I_N')$ is called its calculation using the formula $C = r\mathrm{B}(\tilde{I}')$ provided B is the adder and r is a threshold decision rule.

The general scheme of collective synthesis is as follows. First, the adder B is applied to the element $\tilde{I}' = (I_1', I_2', \ldots, I_N')$. Then the decision rule is applied to the result.

Note that the total number of options will be$(l!)^N$.

Let s be the operator performing the permutation of the columns of matrices $m \times l$ using substitution $\langle j_1, j_2, \ldots, j_l \rangle$ and $S = \{s\}$ be the set of all operators.

Believe that $rs = sr, \forall s \in S$. We continue $s \in S$ to the n-dimensional case $\sigma(\tilde{I}') = (s(I'_1), s(I'_2), \ldots, s(I'_n))$. Denote $\Sigma = \{\sigma\}$, where σ is the continuation of s.

From the definition of the adder, it follows that $\sigma \mathrm{B} = \mathrm{B}\sigma, \forall \sigma \in \Sigma$.

Next, $\forall \tilde{I}' \in \mathrm{I}, \forall \sigma \in \Sigma$ we have $r\mathrm{B}(\sigma(\tilde{I}')) = r\sigma(\mathrm{B}(\tilde{I}')) = s(r\mathrm{B}(\tilde{I}'))$, and finally

$$\{\sigma(\tilde{I}'), \sigma \in \Sigma\} \xrightarrow{r\mathrm{B}} \{s(r\mathrm{B}(\tilde{I}')), s \in S\} = \mathrm{K}(r\mathrm{B}(\tilde{I}')) = \mathrm{K}(\|c_{ij}\|_{m\times l}).$$

So, the product $r\mathrm{B}$ sets the desired mapping and sets some collective clustering.

It is necessary to find the optimal element of K_c and well then $\tilde{I}'$.

$$\mathrm{I} \xrightarrow{r\mathrm{B}} \mathrm{K}_c, \quad A^c_{\tilde{I}'}(X) = \mathrm{K}(r\mathrm{B}(\tilde{I}')).$$

Note that all information about the collective decision is contained in the matrix $B = \mathrm{B}(\tilde{I}')$. We introduce the definitions of potentially the best and worst possible solutions. As an "ideal" of a collective solution, we consider the case when all the algorithms give us essentially the same partitions or covers.

Definition 12.6. The numerical matrix $\|b_{ij}\|_{m\times l}$ is called contrast if $b_{ij} \in \{0, N\}$. A numeric matrix $\|b_{ij}\|_{m\times l}$ is called diffused if $b_{ij} = \delta_i \in R^N$.

As a distance between two numerical matrices, we consider a known function $\rho(B^1, B^2) = \sum_{i=1}^{m} \sum_{j=1}^{l} |b^1_{ij} - b^2_{ij}|$.

Denote by M the set of all contrast matrices and by $\tilde{\mathrm{M}}$ the set of all diffuse matrices. We introduce definitions for assessing the quality of matrices B.

Definition 12.7.

$$\Phi(B) = \rho(B, \mathrm{M}) \underset{B}{\longrightarrow} \min. \tag{12.1}$$

Definition 12.8.

$$\tilde{\Phi}(B) = \rho(B, \tilde{\mathrm{M}}) \underset{B}{\longrightarrow} \max. \tag{12.2}$$

The set $\tilde{\mathrm{M}}' = \{\tilde{B}\}$, where $\tilde{B} = \|\tilde{b}_{ij}\|_{m\times l}, \tilde{b}_{ij} = \frac{N}{2}$, is called the average diffuse matrix. This set is called the average diffuse matrix.

Definition 12.9.

$$\tilde{\Phi}'(B) = \rho(B, \tilde{B}) \underset{B}{\longrightarrow} \max. \tag{12.3}$$

Note that the optimum according to criteria (12.1) and (12.2) do not have to match, but the sets M and $\tilde{\mathrm{M}}$ intersect.

Theorem 12.1. *The sets of optimal solutions by criteria* (12.1) *and* (12.3) *coincide.*

Proof. We show that $\Phi(B) + \tilde{\Phi}'(B) = \frac{Nml}{2}$ for any B.

Write $\tilde{\Phi}'(B) = \sum_{i=1}^{m}\sum_{j=1}^{l}\tilde{\alpha}_{ij}, \tilde{\alpha}_{ij} = |b_{ij} - \frac{N}{2}|, \Phi(B) = \sum_{i=1}^{m}\sum_{j=1}^{l}\alpha^*_{ij}, \alpha^*_{ij} = \min(b_{ij}, N - b_{ij})$.

If $b_{ij} \geq \frac{N}{2}$, then $\tilde{\alpha}_{ij} = b_{ij} - \frac{N}{2}, \alpha^*_{ij} = N - b_{ij}$, and $\tilde{\alpha}_{ij} + \alpha^*_{ij} = \frac{N}{2}$.

If $b_{ij} < \frac{N}{2}$, then $\tilde{\alpha}_{ij} = \frac{N}{2} - b_{ij}, \alpha^*_{ij} = b_{ij}$, and $\tilde{\alpha}_{ij} + \alpha^*_{ij} = \frac{N}{2}$.

Summing over the whole set of values of pairs of indices i, j, we get that $\Phi(B) + \tilde{\Phi}'(B) = \frac{Nml}{2}$. □

Consider the problem of finding optimal collective clusterings for criterion (13.1). It's clear that $\Phi(B) = \sum_{i=1}^{m}\sum_{j=1}^{l}\min(b_{ij}, N - b_{ij})$.

We introduce the notations $M = \{1, 2, \ldots, m\}, X_j = \left\{i | b_{ij} \geq \frac{N}{2}, i = 1, 2, \ldots, m\right\}, Y_j = M \backslash X_j, j = 1, 2, \ldots, l$.

Let $\pi^\nu =< \mu^\nu_1, \mu^\nu_2, \ldots, \mu^\nu_l >, \nu = 1, 2, \ldots, N$

be some permutation of a set $\pi^0 = \langle 1, 2, \ldots, l\rangle, \pi = \langle \pi^1, \pi^2, \ldots, \pi^N\rangle$ be a set of permutations that uniquely defines a rating matrix B', and $B' = \|b'_{ij}\|_{m\times l}, b'_{ij} = b_{ij}(\pi) = \sum_{\nu=1}^{N}\alpha'^{\nu}_{ij}$.

We further assume that $\|\alpha^\nu_{ij}\|_{m\times l}$ is the initial matrix of the algorithm A^c_ν corresponding to the permutation π^0 and $\|\alpha'^{\nu}_{ij}\|_{m\times l}$ is the matrix of the algorithm A^c_ν corresponding to a certain permutation π^ν. Then $\alpha'^{\nu}_{ij} = \alpha^\nu_{i\mu^\nu_j}$. Set $\tilde{\Delta}_\nu = \sum_{j=1}^{l}(\sum_{i\in X_j}\bar{\alpha}^\nu_{ij} + \sum_{i\in Y_j}\alpha^\nu_{ij})$, $\tilde{\Delta}'_\nu = \sum_{j=1}^{l}(\sum_{i\in X_j}\bar{\alpha}'^{\nu}_{ij} + \sum_{i\in Y_j}\alpha'^{\nu}_{ij})$.

So, $\Delta_\nu = \tilde{\Delta}'_\nu - \tilde{\Delta}_\nu = \sum_{j=1}^{l}(\sum_{i\in X_j}(\alpha^\nu_{ij} - \alpha'^{\nu}_{ij}) + \sum_{i\in Y_j}(\alpha'^{\nu}_{ij} - \alpha^\nu_{ij}))$. Convert this expression. Rightly identity $\sum_{j=1}^{l}(\sum_{i\in X_j}\alpha^\nu_{ij} + \sum_{i\in Y_j}\alpha^\nu_{ij}) = \sum_{j=1}^{l}(\sum_{i\in X_j}\alpha^\nu_{i\mu^\nu_j} + \sum_{i\in Y_j}\alpha^\nu_{i\mu^\nu_j})$. Grouping all the

intervals of summation, we get $\Delta_\nu = \sum_{j=1}^{l} (\sum_{i\in X_j} (\alpha_{ij}^{\nu} - \alpha_{i\mu_j^\nu}^{\nu}) + \sum_{i\in Y_j} (\alpha_{i\mu_j^\nu}^{\nu} - \alpha_{ij}^{\nu})) = 2\sum_{j=1}^{l} \sum_{i\in X_j} (\alpha_{ij}^{\nu} - \alpha_{i\mu_j^\nu}^{\nu}) = 2\sum_{j=1}^{l} \sum_{i\in X_j} \alpha_{ij}^{\nu} - 2\sum_{j=1}^{l} \sum_{i\in X_j} \alpha_{i\mu_j^\nu}^{\nu}$.

Thus, minimizing a function Δ_ν is equivalent to maximizing the second sum of a given expression.

After applying permutations $\pi = \langle \pi^1, \pi^2, \dots, \pi^N \rangle$, the set X_j, $Y_j, j = 1, 2, \dots, l$ will change.

We introduce the next notations:

$$M_{1j} = X_j \backslash (Y_j' \backslash Y_j),$$
$$M_{2j} = Y_j' \backslash Y_j,$$
$$M_{3j} = Y_j \backslash (X_j' \backslash X_j),$$
$$M_{4j} = X_j' \backslash X_j.$$

Theorem 12.2.

$$\Delta\Phi = \Phi(B') - \Phi(B) \le \sum_{\nu=1}^{N} \Delta_\nu + \sum_{\nu=1}^{N} \left(|M_{2j}| \begin{cases} -2, & N - even, \\ -1, & N - odd \end{cases} + |M_{4j}| \begin{cases} 0, & N - even, \\ -1, & N - odd \end{cases} \right).$$

Since the second sum is always non-positive, we have the upper bound $\Delta\Phi \le \sum_{\nu=1}^{N} \Delta_\nu$. Consider the function Δ_ν minimization problem. Let's write down all possible variants of the function $\sum_{j=1}^{l} \sum_{i\in X_j} \alpha_{i\mu_j^\nu}^{\nu}$ in the form of a table. Then the minimum of this function is reduced to finding the maximum matching of the bipartite graph, for finding which you can use the polynomial Hungarian algorithm [6].

It's clear that $\min_{\pi^\nu} \Delta_\nu \le 0$. Now we can propose the following heuristic algorithm of the "fastest descent".

Proposed Algorithm.

We describe the optimization step:

1. We calculate $X_j, j = 1, 2, \dots, l$.
2. Find for each ν value $\Delta_\nu^* = \min_{\pi^\nu} \Delta_\nu$.

If $\sum_{\nu=1}^{N} \Delta_\nu^* < 0$, then we apply the found permutations $\pi^\nu =< \mu_1^\nu, \mu_2^\nu, \ldots, \mu_l^\nu >, \nu = 1, 2, \ldots, N$, and proceed to step 1.

If $\sum_{\nu=1}^{N} \Delta_\nu^* = 0$, then the END of algorithm.

Note. Note that our algorithm does not find even the local minimum of the criterion $\Phi(B)$. However, this algorithm is very fast, and its complexity at each iteration is estimated as $O(l^5 mN)$.

Let us have the results of clustering by N algorithms of sampling from m objects into l clusters. We can write solutions in the form of a binary three-dimensional matrix $\|\alpha_{ij}^v\|, \nu = 1, 2, \ldots, N, i = 1, 2, \ldots, m, j = 1, 2, \ldots, l$. We assume that the cluster numbers in each algorithm are fixed. Then any horizontal $i-$layer of a given three-dimensional matrix will denote the results of clustering the object $\mathbf{x}_i$. As a collective clustering of the sample X, we can take the result of clustering of "new" descriptions — the layers of the original matrix $\|\alpha_{ij}^v\|$.

As a collective clustering method, you can take in fact any clustering algorithm. For example, the "k-intragroup averages" method. Let there be a lot $\|\alpha_{ij}^v\|$ of N clusters obtained by heuristic clustering algorithms, and there are t elements $\|\alpha_{i_1 j}^v\|, \|\alpha_{i_2 j}^v\|, \ldots, \|\alpha_{i_t j}^v\|$ of a cluster of collective solutions. Then their sample mean $\|\alpha_j^{*\nu}\|$ is calculated as $\alpha_j^{*v} = \frac{1}{t} \sum_{\mu=1}^{t} \alpha_{i_\mu j}^v$. Note that the "$k$-intragroup averages" method makes it possible to calculate such collective clustering $\mathrm{K} = \{K_1^*, K_2^*, \ldots, K_l^*\}$ that the sets of heuristic clusterings of objects of a cluster of a collective solution will be close to each other in the Euclidean metric.

Committee synthesis of collective decisions gives more interpretable solutions. Indeed, if they are separate solutions $\mathrm{K}^\nu = \{K_1^\nu, K_2^\nu, \ldots, K_l^\nu\}, \nu = 1, 2, \ldots, l$ of heuristic clustering algorithms, a cluster of a collective solution will be the "crossing" of many clusters $K_{i_1}^1, K_{i_2}^2, \ldots, K_{i_N}^N$.

12.3 Restoring Functions Based on a Set of Solutions of Supervised Classification Problems

Consider the task of restoring the function $y = f(\mathbf{x})$ on the training sample $y_i, \mathbf{x}_i, i = 1, 2, \ldots, m$ at $y \in (a, b)$, where $a, b \in R$. This problem is usually in the statistical formulation known as

the problem of regression restoring, i.e. conditional expectation functions, assuming the existence of conditional density $p(y|\mathbf{x})$. Here there are various non-parametric and parametric approaches. Non-parametric approaches are well described in the book [5]. Usually accepted $y = \frac{\sum_{i=1}^{m} w_i(\mathbf{x}) y_i}{\sum_{i=1}^{m} w_i(\mathbf{x})}, w_i(\mathbf{x}) = \Psi\left(\frac{\rho(\mathbf{x},\mathbf{x}_i)}{h}\right)$. Here Ψ is the nuclear function, and h is the width of the window. Linear parametric methods $f(\mathbf{w},\mathbf{x}) = \sum_{j=1}^{k} w_j x_j + w_0$ [4] and their generalizations as logistic $f(\mathbf{w},\mathbf{x}) = \frac{1}{1+e^{-z}}, z = \sum_{j=1}^{k} w_j x_j + w_0$, curvilinear $f(\mathbf{w},\mathbf{x}) = \sum_{j=1}^{n} w_j \phi_j(x_1,\ldots,x_k), \phi_1,\ldots,\phi_n : R^k \Rightarrow R$ and polynomial regressions $f(\mathbf{w},\mathbf{x}) = \sum_{p_1+p_2+\ldots+p_k=\gamma} w_{p_1 p_2 \ldots p_k} x_1^{p_1} x_2^{p_2} \ldots x_k^{p_k}$ are based on the application of the maximum likelihood method and the method of least squares. Let us consider two methods of recovering the dependence $y = f(\mathbf{x})$ on a training sample, based on solving auxiliary supervised classification problems. The general idea is to set the number of tasks N. Each task consists of forecasting the segment to which y belongs and building a collective solution for these N forecasts. We denote by m' the number of different values of the target feature y in the training sample. Consider splitting the domain of valid values into intervals. Consider splitting the domain $\boldsymbol{\Delta} = \{\Delta_1, \Delta_2, \ldots, \Delta_n\}$ of valid values y into intervals $\Delta_1 = (d_0, d_1], \Delta_2 = (d_1, d_2], \ldots, \Delta_n = (d_{n-1}, d_n]$. We can assume that $y_i \le y_{i+1}, i = 1, 2, \ldots, m_n - 1, m_n = m$ and $y_1, y_2, \ldots, y_{m_1} \in \Delta_1, y_{m_1+1}, y_{m_1+2}, \ldots, y_{m_2} \in \Delta_2, \ldots, y_{m_{n-1}+1}, y_{m_{n-1}+2}, \ldots, y_{m_n} \in \Delta_n$. Take an integer L, $2 \le L \le n$, and define N partitions M into L intervals, assigning an integer vector $\mathbf{k}_i = (0, k_i^1, k_i^2, \ldots, k_i^{L-1}, n), i = 1, 2, \ldots, N$, $k_i^j < k_i^{j+1} < n$ to each partition. Each given partition builds a partition of the entire set of objects $\mathbf{x}$ into L disjoint classes according to the rule: $\mathbf{x} \in K_j^i$ of partition $\mathbf{K}^i = \{K_1^i, K_2^i, \ldots, K_L^i\}$ if and only if $y = f(\mathbf{x}) \in \Delta_k$ and $k \in (k_i^j, k_i^{j+1}]$. With this partition $\mathbf{K}^i = \{K_1^i, K_2^i, \ldots, K_L^i\}$, a classification algorithm $A_i, i = 1, 2, \ldots, N$ is constructed. The range of allowable values of the dependent quantity y is divided into n disjoint intervals $\boldsymbol{\Delta} = \{\Delta_i\}_{i=1}^n$ with an error $Q(\boldsymbol{\Delta}) = \sum_{k=1}^{n} \sum_{y_i \in \Delta_k} (y_i - c_k)^2, c_k \in \Delta_k, r = 1, 2, \ldots, n$. At a fixed $\boldsymbol{\Delta}$, minimum of $Q(\boldsymbol{\Delta})$ is reached at $c_k = \frac{\sum_{y_i \in \Delta_k} y_i}{|\{y_i : y_i \in \Delta_k\}|}$. Thus, if the

number of segments is known, then the points c_k ("representatives" of the segments) and points $y_{m_\nu}, \nu = 1, 2, \ldots, n, y_{m_{k-1}} < y_i \leq y_{m_k}, m_n = m, k = 1, 2, \ldots, n, i = 1, 2, \ldots, m$ are found efficiently.

Denote by $\mathbf{K}^1 \times \mathbf{K}^2 \times \ldots \times \mathbf{K}^N \times \boldsymbol{\Delta}$ the set of elementary events. An event $(K^1_{a_1}, K^2_{a_2}, \ldots, K^N_{a_N}, \Delta_k)$ means that an object $\mathbf{x}$ is classified by the algorithm A_i as $\mathbf{x} \in K^i_{a_i}$ for which $y = f(\mathbf{x}) \in \Delta_k$. We denote the probability of this event as $\mathrm{P}(K^1_{a_1}, K^2_{a_2}, \ldots, K^N_{a_N}, \Delta_k)$ or easier $\mathrm{P}(\mathbf{x}, \Delta_k)$. Then

$$\mathrm{P}(\Delta_k | K^1_{a_1}, K^2_{a_2}, \ldots, K^N_{a_N}) = \frac{\mathrm{P}(\Delta_k)}{\mathrm{P}(K^1_{a_1}, K^2_{a_2}, \ldots, K^N_{a_N})} \mathrm{P}(K^1_{a_1}, K^2_{a_2}, \ldots, K^N_{a_N} | \Delta_k). \quad (12.4)$$

With statistically independent classifiers, i.e. "naive" Bayesian approach, we have $\mathrm{P}(K^1_{a_1}, K^2_{a_2}, \ldots, K^N_{a_N} | \Delta_k) = \prod_{i=1}^N P(K^i_{a_i} | \Delta_k)$, $\mathrm{P}(K^1_{a_1}, K^2_{a_2}, \ldots, K^N_{a_N}) = \prod_{i=1}^N \mathrm{P}(K^i_{a_i})$, and (12.4) takes form (12.5)

$$\mathrm{P}(\Delta_k | K^1_{a_1}, K^2_{a_2}, \ldots, K^N_{a_N}) = \frac{\mathrm{P}(\Delta_k)}{\prod_{i=1}^N \mathrm{P}(K^i_{a_i})} \prod_{i=1}^N P(K^i_{a_i} | \Delta_k). \quad (12.5)$$

Definition 12.10. Bayesian corrector is a function F :

$$(\mathrm{P}(\Delta_1|\mathbf{x}), \mathrm{P}(\Delta_2|\mathbf{x}), \ldots, \mathrm{P}(\Delta_n|\mathbf{x})) \to R.$$

Definition 12.11. The answer on the average of the Bayesian corrector for an object $\mathbf{x}$ is the value $\tilde{y} = \sum_{k=1}^n \mathrm{P}(\Delta_k|\mathbf{x}) c_k$.

Definition 12.12. The answer to the maximum Bayesian corrector for the object $\mathbf{x}$ is the value $\hat{y} = c_k$, where $k = \arg \max_{j=1,2,\ldots,n} \mathrm{P}(\Delta_j|\mathbf{x})$.

The general algorithm for recovering dependencies consists of partitioning the set M into intervals $\Delta_k, k = 1, 2, \ldots, n$ and choosing the number of classes L, defining partitions $\mathbf{K}^i, i = 1, 2, \ldots, N$, forming learning samples of recognition tasks $Z_i, i = 1, 2, \ldots, N$, choosing a classification model, and constructing the best classifier

algorithms in models $A_i, i = 1, 2, \ldots, N$, calculating probability estimates $\mathrm{P}(K_j^i|\Delta_k), \mathrm{P}(K_j^i), \mathrm{P}(\Delta_k), i = 1, 2, \ldots, N, j = 1, 2, \ldots, L$, $k = 1, 2, \ldots, n$.

The calculation of the value of the dependent quantity for an arbitrary $\mathbf{x}$ consists of its classification by algorithms $A_i, i = 1, 2, \ldots, N$, calculating the values of conditional probabilities $\mathrm{P}(\Delta_k|\mathbf{x}), k = 1$, $2, \ldots, n$ using formulas (5), calculating the final score as a "response by average" $\tilde{y}$, or as a "response by maximum" $\hat{y}$.

To calculate the parameters of the model of dependency recovery, the following optimization problem is posed:

$$\sum_{i=1}^{m} (y_i - \tilde{y}_i)^2 \to \min_{\mathrm{P}(K_j^i|\Delta_k), \mathrm{P}(K_j^i), \mathrm{P}(\Delta_k)},$$

under restrictions

$$\sum_{j=1}^{L} \mathrm{P}(K_j^i) = 1, \quad i = 1, 2, \ldots, N,$$

$$\sum_{k=1}^{n} \mathrm{P}(\Delta_k) = 1, \mathrm{P}(\Delta_k) > 0, \quad k = 1, 2, \ldots, n,$$

$$\sum_{k=1}^{n} \mathrm{P}(K_j^i|\Delta_k)\mathrm{P}(\Delta_k) = \mathrm{P}(K_j^i), \quad \mathrm{P}(K_j^i|\Delta_k) \geq 0,$$

$$i = 1, 2, \ldots, N, \quad j = 1, 2, \ldots, L.$$

The analytical solution of this problem for $L = 2, N = n - 1$ and fixed values of the interval probabilities $\mathrm{P}(\Delta_k) = \alpha_k, k = 1, 2, \ldots, n$, are

$$\mathrm{P}(K_1^i|\Delta_k) = \begin{cases} 0, & i < k, \\ 1, & i = k, \\ \frac{\alpha_i}{1 - \sum_{t=1}^{i-1} \alpha_i}, & i > k, \end{cases}$$

$$\mathrm{P}(K_2^i|\Delta_k) = 1 - \mathrm{P}(K_1^i|\Delta_k),$$

$$\mathrm{P}(K_1^i) = \frac{\alpha_i}{1 - \sum_{t=1}^{i-1} \alpha_i},$$

$$\mathrm{P}(K_2^i) = 1 - \mathrm{P}(K_1^i), \quad i = 1, 2, \ldots, N, \quad k = 1, 2, \ldots, n. \tag{12.6}$$

Definition 12.13. The model has the property **A** if $\mathrm{P}(\Delta_k|\mathbf{x}_i) \neq 0 \Leftrightarrow y_i \in \Delta_k$ is satisfied for the learning object $\mathbf{x}_i, i = 1, 2, \ldots, m$.

In [14–16], it is shown that the Bayesian corrector with probability estimates (12.6) has the property **A**. An alternative approach to calculating probability estimates for a dependency recovery model is to use frequency estimates for $\mathrm{P}(K_1^i|\Delta_k)$ and $\mathrm{P}(K_2^i|\Delta_k)$. However, in practical construction $\mathrm{P}(K_j^i|\Delta_k)$, situations arise when classifiers attribute an object $\mathbf{x}$ to classes in such a way that $\mathrm{P}(K_j^i|\Delta_k) = 0$, $k = 1, 2, \ldots, n$. Consider an approach in which you can avoid such situations:

$$\tilde{\mathrm{P}}(K_j^i|\Delta_k) = \frac{\sum_{t=\max\{-d,1-k\}}^{t=\min\{d,n-k\}} w_t \mathrm{P}(K_j^i|\Delta_{k+t})}{\sum_{t=\max\{-d,1-k\}}^{t=\min\{d,n-k\}} w_t},$$

$$i = 1, 2, \ldots, n, \quad j = 1, 2, \ldots, l, \quad k = 1, 2, \ldots, n,$$

$$\tilde{\mathrm{P}}(K_j^i) = \sum_{k=1}^{n} \tilde{\mathrm{P}}(K_j^i|\Delta_k)\mathrm{P}(\Delta_k).$$

Values $\tilde{\mathrm{P}}(K_j^i|\Delta_k)$ are formally probabilities. We will call the value d as the width of the smoothing window, and the magnitudes $w_{-d}, \ldots, w_d$ are the smoothing weights. The replacement process $\mathrm{P}(K_j^i|\Delta_k) \to \tilde{\mathrm{P}}(K_j^i|\Delta_k)$ is called smoothing.

There are restrictions on smoothing weights:

- non-negativity $w_t \geq 0, t = -d, \ldots, d$,
- symmetry $w_t = w_{-t}, t = 1, 2, \ldots, d$,
- character of decreasing $w_t = u^{d-|t|}, u > 1, t = -d, \ldots, d$.

Definition 12.14. A model has a property **B** if $y_i \in \Delta_k \Rightarrow \mathrm{P}(\Delta_k|\mathbf{x}_i) > \mathrm{P}(\Delta_{k_1}|\mathbf{x}_i), k \neq k_1$ is performed for the learning object $\mathbf{x}_i, i = 1, 2, \ldots, m$. $y_i \in \Delta_k \Rightarrow \mathrm{P}(\Delta_k|\mathbf{x}_i) > \mathrm{P}(\Delta_{k_1}|\mathbf{x}_i), k \neq k_1$.

It is shown that the Bayesian corrector with smoothed frequency estimates of probabilities and the "maximum answer" has the property **B**.

Definition 12.15. A dependency recovery model is said to be quasi-correct with respect to partitioning $\boldsymbol{\Delta}$ if $y_i \in \Delta_k \Rightarrow \hat{y}_i \in \Delta_k, i = 1, 2, \ldots, m$ is satisfied for the training set $y_i, \mathbf{x}_i, i = 1, 2, \ldots, m$.

A model is called correct if $\tilde{y}_i = y_i, i = 1, 2, \ldots, m$ is satisfied for given $y_i, \mathbf{x}_i, i = 1, 2, \ldots, m$.

Definition 12.16. The model $\mathbf{A}_1$ is the Bayesian corrector over the correct classifiers with analytical estimates of the probabilities when using the "answer by average".

Definition 12.17. The model $\mathbf{A}_2$ is the Bayesian corrector over the correct classifiers with frequency estimates of the probabilities when using the "answer to the maximum".

Definition 12.18. The model $\mathbf{A}_2^d$ is the Bayesian corrector over the correct classifiers with smoothed frequency estimates of the probabilities and the width d of the smoothing window when using the "answer to the maximum".

Theorem 12.3. *Models* $\mathbf{A}_1, \mathbf{A}_2$, *and* $\mathbf{A}_2^d$ *with consistent training information are quasi-correct with respect to partitioning* $\boldsymbol{\Delta}$. *When* $n = m'$, *the ranges of values of the target characteristic of these models are correct.*

In this section, we consider the "linear corrector" and its properties with respect to the partitioning of the target attribute. We introduce the notation:

$$u_i^{(j)} = \frac{\sum_{k=k_i^j+1}^{k_i^{j+1}} c_k}{k_i^{j+1} - k_i^j}, \quad i = 1, 2, \ldots, N, \quad j = 1, 2, \ldots, L.$$

Let's make a vector of answers $\mathbf{u}(\mathbf{x}) = (1, u_1(\mathbf{x}), u_2(\mathbf{x}), \ldots, u_N(\mathbf{x}))$, $u_i(\mathbf{x}) = u_i^{(j)}$ if the classifier $A_i, i = 1, 2, \ldots, N$, took the object $\mathbf{x}$ to the class K_j^i.

Definition 12.19. A linear corrector is a function $f : X \to R$, $f(\mathbf{x}) = (\mathbf{u}(\mathbf{x}), \mathbf{w})$, where $\mathbf{w}.\mathbf{w} \in R^{N+1}$ is the weight vector.

The weights vector is the solution to the optimization problem $\sum_{i=1}^m (y_i - f(\mathbf{x}_i))^2 \to \min_{\mathbf{w}}$.

The algorithm for restoring dependency by a linear corrector consists of splitting the set M into intervals $\Delta_k, k = 1, 2, \ldots, n$ and the number of classes L, defining the splits $\mathbf{K}^i, i = 1, 2, \ldots, N$, generating training samples of recognition tasks $Z_i, i = 1, 2, \ldots, N$, choosing a classification model, and calculating weights $\mathbf{w}$.

The algorithm for calculating the value of the dependent variable consists of the classification $\mathbf{x}$ by the algorithms $A_i, i = 1, 2, \ldots, N$, calculating the response vector $\mathbf{u}(\mathbf{x})$, and the value of the function $f(\mathbf{x})$.

Theorem 12.4. *Linear corrector when $n = m'$ is the correct model.*

For the case $L = 2, N = n - 1, n = m'$, it can be shown that the weight vector has the form $w_k = \frac{y_k - y_{k-1}}{u_k^{(2)} - u_k^{(1)}}, k = 2, 3, \ldots, n, w_1 = y_1 - \sum_{k=2}^{n} u_k^{(1)} w_k$.

Numerical comparisons of the created correct methods with a linear corrector and nuclear smoothing on a one-dimensional parabola, sinusoid, two-dimensional Gaussian, and a number of practical problems of small dimensions are carried out. The results of the presented Bayesian and linear corrector were generally significantly higher.

References

[1] Aggarwal, C. (2015) *Data Mining: The Textbook* (IBM T.J. Watson Research Center. Yorktown Heights, New York), 771 p. DOI: 10.1007/978-3-319-14142-8.

[2] Biryukov, A.S., Shmakov, A.S., and Ryazanov, V.V. (2008) Solution of cluster analysis problems by groups of algorithms. *Journal of Computational Mathematics and Mathematical Physics*, 48(1), 176–192.

[3] Desgraupes, B. (2013) Clustering Indices. University Paris Ouest. Lab Modal'X.

[4] Draper, N. and Smith, G. (1998) *Applied Regression Analysis* (John Wiley & Sons Inc, New York), 736 p.

[5] Hardle, V. (1993) *Applied Nonparametric Regression* (Mir Publishers, Moscow), 349 p.

[6] Kuhn, H. (1955) The Hungarian method for the assignment problem. *Naval Research Logistics Quarterly*, 2, 83–97. DOI: 10.1002/nav.3800020109.

[7] Kuncheva, L. (2014) *Combining Pattern Classifiers: Methods and Algorithms* (John Wiley & Sons Inc), 360 p., DOI: 10.1002/9781118914564.

[8] Method of committees in pattern recognition. (1974) Sverdlovsk, IMM AN USSR.

[9] Rastrigin, L. and Erenstein, R. (1975) Collective decision-making in pattern recognition. *Avtomatica i telemehanika*, 9, 133–144.

[10] Ryazanov, V.V. (1981) Committee synthesis of recognition and classification algorithms. *Zhurnal Vychislitel'noi Matematiki i Matematicheskoi Fiziki*, 21(6), 1533–1543.
[11] Ryazanov, V.V. (1982) The synthesis of classifying algorithms on finite sets of classification (taxonomy) algorithms. *Zhurnal Vychislitel'noi Matematiki i Matematicheskoi Fiziki*, 22(2), 429–440.
[12] Ryazanov, V.V. (1995) One approach for classification (taxonomy) problem solution by sets of heuristic algorithms. *Proceedings of the 9th SCIA*, Uppsala, Sweden, June 1995, pp. 997–1002.
[13] Ryazanov, V.V. (2007) Logical regularities in pattern recognition problems (parametric approach). *Zhurnal Vychislitel'noi Matematiki i Matematicheskoi Fiziki*, 47(10), 1793–1808 (*Computational Mathematics and Mathematical Physics*, 47(10), 1720–1735, October 2007).
[14] Ryazanov, V.V. and Tkachev, Y.I. (2010) The solution of the problem of restoring dependence teams of recognition algorithms. *Reports of the Academy of Sciences*, 432(6), 750–754.
[15] Ryazanov, V.V. and Tkachev, Y.I. (2010) Dependency recovery based on Bayesian correction of collectives of classification algorithms. *Journal of Computational Mathematics and Mathematical Physics*, 50(9).
[16] Tkachev, Y.I. (2013) Methods for solving the problem of restoration of dependence by collectives of recognition algorithms. PhD thesis, CC RAN, Moscow.
[17] Zhuravlev, Y. (1977, 1977, 1978) Correct algebras over sets of noncorrect (heuristic) algorithms. I. *Cybernetics*, 4, 5–17. II. *Cybernetics*, 6, III. *Cybernetics*, 2, 35–43.
[18] Zhuravlev, Y. (1978) On the algebraic approach to solving problems of recognition or classification. *Problems of Cybernetics*, 33, 5–68.
[19] Zhuravlev, Y. (1998) *Selected Scientific Works* (Publishing House Magister, Moscow), 420 p.
[20] Zhuravlev, Y.I., Ryazanov, Y.I., and Sen'ko, O.V. (2006) *Recognition. Mathematical Methods. The Software System. Practical Applications* (PHASIS, Moscow), p. 168.

© 2025 World Scientific Publishing Company
https://doi.org/10.1142/9789811267215_0013

Chapter 13

Solving High-Dimensional Inverse Problems with Artificial Neural Networks

Sergey A. Dolenko[*,§], Igor V. Isaev[*,¶], Alexander O. Efitorov[*], Sergey A. Burikov[*,†], Tatiana A. Dolenko[*,†], Eugeny A. Obornev[‡], and Mikhail I. Shimelevich[‡]

[*] *D.V. Skobeltsyn Institute of Nuclear Physics, M.V. Lomonosov Moscow State University, Moscow, the Russian Federation*

[†] *Physics Department, M.V. Lomonosov Moscow State University, Moscow, the Russian Federation*

[‡] *S. Ordjonikidze Russian State Geological Prospecting University, Moscow, the Russian Federation*

[§] *dolenko@srd.sinp.msu.ru*

[¶] *isaev_igor@mail.ru*

Inverse problems (IP) represent a very common class of problems that arise when processing experimental data. In fact, most problems of indirect measurement, used to determine some characteristics of the studied object, belong to this class. Unfortunately, many IP are incorrect or ill-conditioned, i.e. their solution is unstable or not unique. These properties greatly complicate the solution of such problems, and in the case of problems high-dimensional by input (measured data) and/or by output (object parameters) they make it virtually impossible, if the methods used are traditional mathematical methods based on operator matrix approach or on optimization algorithms.

The alternative is to use adaptive nonlinear approximation methods, in particular, artificial neural networks (ANN). Thanks to their

properties, such as resilience to noise and inconsistent data, and training by examples, ANN with the correct method of their application are able to carry out adaptive regularization of ill-posed problems, and often they are one of the most effective methods of their solution.

This chapter discusses the methodological aspects of the solution of IP with ANN. Various statements of IP from the point of view of used methods of data processing are given. Various methodological approaches to the solution of the IP when using neural network methods are considered, as well as their properties, differences, and application areas. The differences between ANN and other methods of IP solution are discussed, as well as the main areas where their use is justified. The material is illustrated by high-dimensional IP from the areas of optical spectroscopy and exploration geophysics (electrical prospecting). Various approaches when solving multiparameter IP are discussed: autonomous, simultaneous, group, and stepwise parameter determination.

Keywords: High-dimensional inverse problems, artificial neural networks, optical spectroscopy, exploration geophysics.

13.1 Introduction

The modern stage of scientific development is characterized by the increasing complexity of obtaining information in experimental research. The times when the researcher was able to directly measure the values of his interest, apparently, are gone forever. Today, almost all information is obtained as a result of indirect measurements, which implies the need for subsequent transition from directly measured (observed) values to the parameters of the object of the researcher's interest. The implementation of such a transition requires the construction of mapping of the space of observed values to the space of parameters, which constitutes a solution of a so-called inverse problem (IP).

Unfortunately, in many cases, the solution of the IP turns out to be a very serious problem, which is usually associated with the properties of the mapping itself that needs to be constructed. The IP emerging in practice being nonlinear and ill-posed quite often leads to situations where traditional methods of solving the IP experience difficulties or fail. It is necessary to study the properties of new approaches to the IP solution, to develop methods for their application, and to compare the results with the results of traditional approaches, in order to determine the optimal application areas for both.

Artificial neural networks (ANN), due to their well-known properties [41], especially such as nonlinearity, adaptability, and learning by example, are universal nonlinear approximators, which allows one to use them successfully to solve IP [86]. Due to their resilience to noise and to contradictory input data, ANN are often applicable as well to the solution of ill-posed IP.

This chapter is devoted to the consideration of ANN methods of solving IP and of methodological aspects of their application. It is based on the results obtained by the authors during about 20 years of work in the field of development and application of machine learning methods to solve IP.

This chapter has the following structure.

Section 13.1 (author: S.A. Dolenko) describes the general methodological basis of the solution of IP using ANN (Section 13.1.1 co-authored by M.I. Shimelevich and Sections 13.1.5 and 13.1.6 by I.V. Isaev).

Section 13.2 (authors: S.A. Dolenko, A.O. Efitorov, S.A. Burikov, and T.A. Dolenko) is devoted to the consideration of IP of optical spectroscopy concerning the determination of concentration of each component in multicomponent solutions.

In Section 13.3 (authors: M.I. Shimelevich and E.A. Obornev), the multiparameter IP of electrical prospecting in geophysics is discussed in detail.

13.2 Methodological Basis of the Solution of IP Using ANN

13.2.1 *General statement of the inverse problem*

Consider the studied object of arbitrary nature, whose behavior is determined by the vector of parameters $X = \{x_1, x_2, \ldots, x_n\}$. Let the value of X be unknown, and the behavior of the object is manifested externally as a vector of observed values $Y = \{y_1, y_2, \ldots, y_m\}$. Thus, the studied object implements an unknown function $Y = F(X)$. As a rule, $m \gg n$, i.e. the system can actually be described by a much smaller number of parameters than the number of the observed values. To do so, it is necessary to learn how to restore the X values by Y values. This problem is a problem of modeling the inverse function $X = F^{-1}(Y)$, and it is called an inverse problem (IP).

An IP can be unambiguously solved not always because of possible non-uniqueness and instability of solutions.

Consider, for example, the case when the function $F(X)$ (generally nonlinear) of the direct problem is known, and the inverse problem is reduced to the solution of functional equation of the first kind in some functional, e.g. metric, spaces $\hat{X}$, $\hat{Y}$ [59, 87]:

$$F(X) = Y, \quad X \in \hat{X}, \quad Y \in \hat{Y}. \tag{13.1}$$

The inverse problem (13.1) is called *correct by Hadamard* if we have the following:

(1) For any $Y \in \hat{Y}$, there exists a single solution $X \in \hat{X}$ of the equation (13.1).
(2) The solution X continuously depends on the right side Y, i.e. infinitely small variations of Y (in the metric of the space $\hat{Y}$) correspond to infinitely small variations of the solution X (in $\hat{X}$).

In case of violation of the Hadamard correctness, the inverse problem (13.1) is called *correct by Tikhonov* (conditionally correct), if in the space $\hat{X}$ one can select a subspace $\hat{X}' \subset \hat{X}$ such that we have the following:

(1) It is known *a priori* that a solution X exists and belongs to $\hat{X}'$: $X \in \hat{X}'$.
(2) The solution X is unique.
(3) Infinitely small variations of the solution X correspond to infinitely small variations of the right side of Y, not taking the solution X out of $\hat{X}'$.

In practice, the IP is considered on a bounded closed subset $\hat{X}_N \in R^N$ of a finite-dimensional space R^N (see also Section 13.3). In this case and under the condition that the IP has a unique solution for accurate data, and the function $F(X)$ is continuous and one-to-one, the problem of solving equation (13.1) belongs to the class of conditionally correct (conditionally well-posed) problems.

It should be noted that even with the theoretical correctness of the problem, practical incorrectness may occur due to the presence of noise in the experimental data (which can lead to the loss of resilience of the solution) and due to the discreteness of the set of experimental points (which can lead to non-uniqueness of the solution) — the observed values can be described by several different functions $\boldsymbol{Y} = F_1(\boldsymbol{X})$, $\boldsymbol{Y} = F_2(\boldsymbol{X})$, etc.

As it was noted above, ANN are characterized by high noise resilience and by reliable operation in conditions of complex and contradictory data. This allows one to hope that the formulation of the IP in the form suitable for the application of ANN and the subsequent use of ANN to solve it will improve the practical correctness of the problem, which is of great importance in practice.

13.2.2 *Statements of IP in terms of data processing methods*

From the point of view of data processing methods, an IP can be stated in several different ways:

(1) For the statement of an IP as *an evaluation problem*, let us assume that we have a large dataset consisting of the observed values of $\boldsymbol{Y}$ and of the corresponding solutions of the IP $\boldsymbol{X} = F^{-1}(\boldsymbol{Y})$. At the moment we do not consider the origin (sources) of the information about the $\boldsymbol{X}$ values. This is done in detail in the following.

Availability of such data allows one to use ANN training on these patterns to estimate the values of the sought-for parameters $\boldsymbol{X}$ by the observed values $\boldsymbol{Y}$. The result of such estimation is the solution of the considered IP.

Obviously, besides ANN methods, other methods of restoring dependencies by data samples [89] can be used to solve this problem, but they are beyond the scope of this chapter.

(2) If the researcher has a model for solving the *direct problem*, i.e. a method for calculating the direct function (DF) $\boldsymbol{Y} = F(\boldsymbol{X})$, IP can be stated as an *optimization problem*. In this case, with the help of any method of solving multidimensional optimization problems [36] (e.g., simplex method, gradient methods, or genetic algorithms), the search for such a vector of the required parameters $\boldsymbol{X}$ is carried out, for which the discrepancy between the corresponding calculated vector $\boldsymbol{Y}_{\mathbf{calc}}$ and vector of the observed values $\boldsymbol{Y}_{\mathbf{obs}}$ is minimal: $\| \boldsymbol{Y}_{\mathbf{calc}} - \boldsymbol{Y}_{\mathbf{obs}} \| \to \min$.

Note that the method of calculating the DF $\boldsymbol{Y} = F(\boldsymbol{X})$ can be different. If an adequate analytical model of the DF exists, its values can be simply calculated by the appropriate formula. If an iterative computational model of the DF exists, its values are obtained as the result of appropriate calculations; this option, as a rule, requires significant computational resources and is unfavorable. Finally, it is

possible to build an adaptive parametric model of the DF (ANN model among them), which is then used to calculate the values of the DF in the process of solving the optimization problem. Note that obtaining the solution to the direct problem can turn out to be no less difficult than solving the IP; in this case, such a statement is not suitable.

(3) Note also an important special case when the vector of parameters can take only discrete values. In this case, one can talk about solving the *classification problem* or the *problem of pattern recognition*. In this case, a wide range of specific methods for solving classification problems can be used (see, e.g. [25]).

In this chapter, we focus on the consideration of the ANN solution of IP as an evaluation problem.

13.2.3 *Methodological approaches to IP solution using data-driven methods*

Two fundamentally different basic methodological approaches and one intermediate approach are possible in the practical solution of IP with the help of ANN [38]. Let us describe these approaches and investigate their properties and application areas:

1. "Experiment-based" approach:

An experimental dataset is given: $S_k = (\boldsymbol{X}, \boldsymbol{Y})_k$, $k = 1, \ldots, N$. Here, $\boldsymbol{Y}$ is the vector of the observed values, and the corresponding vector of the required parameters is determined in the experiment by alternative (not neural network) methods. Work with ANN is carried out in two stages:

(a) Division of the entire dataset into training, validation, and test sets. Let us recall that the training dataset is used directly for ANN training, validation set — to determine the moment of termination of training or some additional parameters of ANN, and test set — to estimate the quality of the obtained ANN solution of the problem on independent data.
(b) Training of the ANN on these sets to evaluate the values of the inverse function $\boldsymbol{X} = F^{-1}(\boldsymbol{Y})$.

2. "Model-based" approach:

It is assumed that there is a sufficiently adequate model of the direct function $\boldsymbol{Y} = F(\boldsymbol{X})$. Such a model can be an analytical formula or an algorithm for a computational solution of the direct problem. The work with ANN is carried out in three stages:

(a) Setting the range of change of $\boldsymbol{X}$ and the "grid" of values for each of the parameters of problem $x_1, x_2, \ldots, x_n$.
(b) Obtaining the necessary amount of representative data for all sets (training, validation, and test) by solving the direct problem using the available model of the direct function $\boldsymbol{Y} = F(\boldsymbol{X})$ in the nodes of the "grid".
(c) Training the ANN on these sets to evaluate the values of inverse function $\boldsymbol{X} = F^{-1}(\boldsymbol{Y})$.

A comparative analysis of these two main methodological approaches to the solution of an IP (as well as of an intermediate one, see the following) is given in Table 13.1. Note that if an adequate model is available, the "model-based" approach looks more preferable.

One should separately note the intermediate, so-called "quasi-model" approach, used in the situation when an adequate analytical model of the object is not available due to the high complexity of the object or insufficient information about it. This approach uses an interpolation or extrapolation model based on several "reference" vectors of the observed values obtained experimentally instead of the analytical model based on the existing concepts about the processes occurring in the object.

For example, the vector of the observed values is the spectrum of the object, which depends on one sought-for parameter. If the analytical dependence of the spectrum on the parameter is unknown and the spectrum cannot be calculated numerically, a hypothesis can be put forward about the nature of the dependence of the spectrum on the parameter. In particular, assuming the linear nature of this dependence in each channel of spectrum, it is possible to simulate the spectrum at any value of the parameter, having two "reference" experimental spectra for different values of the parameter and producing linear interpolation and extrapolation.

Table 13.1. A comparison of the properties of the methodical approaches to the solution of an IP with the help of ANN.

Characteristic	"Experiment-based"	"Quasi-model"	"Model-based"
Representativity of datasets	As a rule, insufficient	Adequate	Adequate
Presence of noise in data	Always present	Always present	Can be introduced artificially during calculation of data or ANN training
Level of noise in data	Determined by experiment	Determined by experiment	Set in calculation or training
Conducting experiments	Required, extensive	Required, limited	Desirable, for verification of results
Availability of analytical or computational model for solving the direct problem	Not required	A "quasi-model" required	Required
Adequacy of the solution to the real object	High	To the extent of adequacy of used "quasi-model"	To the extent of adequacy of used model

It is clear that such a "quasi-model" will be good only to the extent that the hypothesis on the nature of the dependence of the vector of the observed values on the desired parameters is valid. As a rule, it is quite simple and can be put forward only for a small number of sought-for parameters, more often for one or two. Note that this hypothesis does not always have a physical justification, and it can be simply a decomposition of the vector of the observed values into a series of the sought-for parameters. Therefore, "quasi-model" is almost always worse than a real analytical model. This disadvantage is partly compensated by the fact that the "reference" vectors of the observed values obtained experimentally are more adequate to the object than the model ones, since they automatically take into account all the features of the influence of the sought-for parameters on the vector of the observed values, even those that for various reasons cannot be taken into account within the "model-based" approach.

Table 13.2. Fields of application of the methodological approaches.

Is an adequate analytical model of the object available?	Yes	No	No
Is a sufficient amount of experimental data available?		Yes	No
Recommended methodological approach	"model-based"	"experiment-based"	"quasi-model"

Comparison of the intermediate approach to the solution of an IP with the two main methodological approaches is carried out in Table 13.1.

Based on all the considerations above, one can make conclusions about the scope of application of the described approaches (Table 13.2).

Note that the boundaries between the fields of application are rather arbitrary because it is not always possible to answer the questions stated in the Table. In practice, two methodological approaches to the solution of a problem can often be applied simultaneously.

13.2.4 *Properties, differences from the other methods, and applications of NN in solving inverse problems*

Let us try to answer the question: What is the fundamental difference between ANN as a method of solving IP and other methods? In other words, what are the advantages of ANN over other algorithms for solving IP, due to what they arise, and when are they revealed?

First, ANN is based in its assessment on information from the entire training database, not just from the specific vector of observed values to which it is currently applied.

Second, the statement of the problem of determination of the sought-for parameters as an optimization problem (solved, for example, by gradient method or genetic algorithm) assumes a fairly accurate knowledge of the nature of the studied dependence (i.e., an

adequate model of the object with a small number of variable parameters, which should be determined in the process of optimization for the given vector of the observed values). To work with this vector, an ANN uses a large number of internal features adaptively allocated in the process of training and does not require accurate knowledge of the model. The model can be used only to create the initial training set (within the "model-based" approach); however, the patterns with the addition of noise presented to the ANN during training are not precisely described by this model anymore. They belong to a more widely understood set of the studied vectors. Due to lack of connection of the sought-for solution with a specific model, the ANN turns out to be more adapted for work in the presence of noise than any methods based on information about the model given *a priori*.

Third, when using conventional methods, an IP is usually solved as an optimization problem, and the optimization procedure uses an indirect measure as a target function to estimate the proximity of the expected values of the parameters to the true values — the distance in the space of the observed values, that is, the proximity (most often in the sense of the standard deviation) of the curve observed in the experiment (for example, a spectrum) to the model curve obtained for the expected values of the parameters. An ANN, on the contrary, uses for this estimation during training the distance in the space of the sought-for parameters, that is, the direct difference between the true values of the parameters and those assumed by the network. Note that although we assume the stability of the solved IP, it does not mean that the minimum of the residual in the space of parameters in the presence of noise will be observed in the point corresponding to the minimum of the residual in the space of the observed values [45]. At the same time, the purpose of the IP solution is the determination of the parameters as precise as possible, rather than the approximation of the vector of observed values as accurately as possible.

All these differences characterize ANN as a data-driven method and determine its unique properties when solving IP of a very different nature. The main properties that are significant for ANN as a method of IP solving are as follows:

- training by example,
- resilience to contradictory data,

- resilience to noise,
- resilience to moderate changes in the side parameters of the object that are not sought in the process of solving the IP.

The listed differences and properties are partially illustrated by the following examples. Extensive experience in ANN solution of IP, accumulated over the past 20 years, allows us to make a very important conclusion that ANN as a method of solving IP expands the capabilities of modern methods of indirect measurements.

The recommended areas of application of ANN as a method of solving IP are also determined by their properties. ANN should be used when

- the IP cannot be solved analytically or numerically,
- data are contradictory or characterized by significant noise levels,
- an analytical or computational model for solving the direct problem is unavailable or unstable,
- *a priori* information about the object is unavailable or difficult to use.

13.2.5 *Various approaches to determination of parameters when solving multiparameter inverse problems*

When solving multiparameter IP (with N_O output parameters), the problem is usually solved separately for each component of the output vector of parameters. Thus, for the complete solution of the IP, it is required to solve N_O subproblems by building N_O ANNs with single output each.

Meanwhile, in the ANN solution of multiparameter IP, there are several possible approaches:

(1) Solution of a separate single-output IP with the construction of a separate ANN for each of the determined parameters, as described above (*autonomous determination*). This approach is the most universal one and is used most often.
(2) Solution of one IP with *simultaneous determination* of all the required parameters, which corresponds to the construction of a single ANN with N_O outputs. The effectiveness of this approach

degrades quickly with the increase in the number of determined parameters. At $N_O > 20$ it becomes inapplicable.

(3) Combining parameters into groups with simultaneous determination of parameters (and construction of one NN) within each group (*group determination*). The method of grouping is dictated by the physical meaning of the determined parameters and known relationships between them. When properly used, this approach may be efficient to reduce solution error in multiparameter IP [19]. Note that we have checked the applicability of this approach for several problems, real world and model, and we conclude that its efficiency is due to the internal properties of the multilayer perceptron as a data processing algorithm, not to the properties of data itself. One more interesting effect of group determination of parameters is its ability to increase the noise resilience of the solution compared to that of autonomous determination [50, 53].

(4) *Step-by-step (sequential) determination* of parameters. In this approach, at the first step, those parameters for which this problem can be solved with acceptable accuracy are determined independently or simultaneously. At the subsequent stages, the values of these parameters obtained by applying the ANN of the first stage are fed to the input of the next ANN along with the values of the input features [44]. This approach is the most difficult one to implement, but sometimes it can improve the quality of the IP solution for those parameters for which the quality of the IP solution in other approaches is unacceptable.

13.2.6 *Noise addition during training as a means to increase noise resilience of the solution of an inverse problem*

As it was already mentioned, one of the peculiarities of IP is the possible instability of their solution. That is why the presence of noise in the input data reduces the efficiency of the methods of IP solution, both traditional and ANN based. In practice, the input data of an IP nearly always contains noise caused by measurement inaccuracy. While ANN itself is resilient to noise, this is often not enough for IP. So, elaboration of the approaches that could increase noise resilience of the methods of IP solution is a very topical problem.

Adding noise at ANN inputs during training is a method that has been known for a long time. In [43, 67], it has been demonstrated that it increased the generalization capabilities of the ANN. In [5], it has been shown that use of this method was equivalent to regularization by Tikhonov. Most often it was used as one of the methods to avoid ANN overtraining [3, 70, 96]. Wang and Principe [90] demonstrated that if this method was used, the ANN also trained faster. Adding noise is also currently used when training deep ANN [94].

The authors of this chapter have previously used this method with ANN solution of IP in laser spectroscopy with a small number of parameters (up to 3) [13, 33, 38]. It has been shown that adding noise during ANN training allowed increasing resilience of the solution against noise in the data; however, the efficiency of this technique heavily depended on the specifics of the particular task being solved.

Testing this approach on various multiparameter IPs during the last 10 years, we confirmed the efficiency of noise addition during ANN training to improve noise resilience of the ANN solution [46–48, 51], for various types of noise. Note that the nature, statistics, and amplitude of the noise added to training data should correspond to those encountered in the test data.

It should be also noted that joint application of group determination of parameters and of training with noise addition to improve the resilience of an ANN solution of an IP sums up the effects of both approaches and makes the solution more noise-resilient than that in the cases when each of the two approaches is applied separately [49, 52].

Now let us consider two examples of inverse problems of high dimensionality, successfully solved by neural network methods.

13.3 Solving Inverse Problems of Laser Spectroscopy Using Artificial Neural Networks

13.3.1 *Review of studies*

Artificial neural networks (ANN) are widely used for the solution of various inverse problems of spectroscopy, primarily for the classification and clusterization of various substances by their

spectra. The first papers devoted to these problems were published in the 1990s [12, 71, 78, 85, 93]. In general, these publications considered the classification of chemical compounds in solid state or biological objects by their IR spectra. Thus, the authors [85] with the help of ANN developed a method of identification of various chemical compounds by 1129 IR spectra from unintegrated large-scale spectral database system (SDBS) of the American National Chemical laboratory for industry. ANN with error backpropagation were used for the classification of different strengths of drugs by their NIR spectra [93]. The authors conducted a comparative analysis of four ways of formation of training set — random selection, Kennard–Stone algorithm, Kohonen self-organizing map, and D-optimal design with a simulated dataset. The obtained results showed that methods based on Kennard–Stone and D-optimal designs are better than the other two and provide 100% correct classification for both recognition and prediction. In the 1990s, methods of classification by FTIR spectra of healthy brain cells and cells affected by Alzheimer's disease were actively developed [71]. Broad prospects for the application of neural networks in spectroscopy became obvious already in the late 1990s [12].

In recent decades, the solution of problems of classification and clusterization of spectral material continues to develop actively. Many authors still work with IR and FTIR spectra of biological objects and drugs [37, 60, 68, 88, 92, 95]. Thus, in [68], it was possible to identify cells belonging to the species Campylobacter coli and Campylobacter jejuni by their IR spectra using multilayer perceptrons (MLP) and probabilistic ANN. The four-layer MLP network correctly identified 99.16% of unknown cells and was successful in detecting atypical patterns from closely related Campylobacter strains and other bacterial species. The problems of recognition of biological objects were successfully solved in [37, 60, 92, 95]. With the help of ANN, the method of identification of 92 species of lactic acid bacteria out of nine genera and strain identification of probiotic and environmental *Enterococcus faecium* strains by FTIR spectra was developed in [92]. These bacteria are characterized by species richness, by a high degree of heterogeneity between species, and by the simultaneous presence of very close species. Use of ANN made it possible to identify the type of bacteria with an accuracy of 93.2%. In the studies [37, 60, 95], problems of diagnostics of ovarian cancer by

means of the analysis of serum or blood plasma [37], different strains of bacteria, and cells [60, 95] were successfully solved.

The authors [26, 61] carried out an interesting work on the use of ANN for the diagnostics of skin cancer by radiation spectra and fluorescence excitation spectra. Total synchronous fluorescence spectra of normal skin, nevus, and melanoma samples were used for training two different types of ANN — the self-organizing map and the feed-forward neural network. ANN responses were compared with the results of histopathology of samples. As a result, both ANN provided high sensitivity with classification errors between 2 and 4%.

Since the 2000s, ANNs are used for the solution of regression problems in vibrational and optical spectroscopy. In [63,91], the method of determination of the concentration of glucose in urine [63] and blood [91] by near IR spectroscopy [63] and FT-Raman spectroscopy [91] using ANN was proposed. The authors [10] used ANN to determine the concentration of N-butylpyridinium chloride in water by fluorescence spectra of solutions. This substance is widely used as an effective catalyst in many chemical reactions. Therefore, precise control of its concentration will provide higher yields of the desired compounds in chemical processes. Use of ANN was caused by the nonlinearity of the measured fluorescence intensities with respect to the compound concentration. It was found that in the concentration range from 72 to 1370 mm ANN allowed a satisfactory assessment of the concentration of the studied substance in aqueous solution with errors of 4.9%, 2.5%, and 1.7% for diode, continuous, and pulsed lasers, respectively. Moreover, the coefficient of multiple determination R^2 was greater than 0.99 in all three cases. The authors [56, 57] have demonstrated the principle possibility of determining toxic impurities — chromium, sulfides, benzene, toluene, naphthalene, etc. — in water by absorption spectra and fluorescence alone or in mixtures using ANN.

The authors of this chapter [15] succeeded in using an MLP to separate the fluorescent signal of oil pollution from the background caused by fluorescence of dissolved organic matter in natural water and to quantify the oil content in the sea. A similar method was used to solve the three-parameter inverse problem of fluorescence spectroscopy for monitoring the excretion of fluorescing medical teranostic nanocomposites (based on carbon quantum dots) and their components from the body with urine [75].

As it is known, Raman spectroscopy is a powerful tool for studying the composition and structure of substances. One of the typical problems of vibrational spectroscopy is to determine the vibrations of which molecular groups contribute to the formation of the Raman or IR absorption spectra. If the object is sufficiently complex (a complex organic molecule or a multicomponent mixture), vibrational bands of different groups overlap, and the problem of spectrum interpretation becomes much more complicated. ANN-based methods in this case can effectively solve this problem. For example, in [35], the problem of determination, which of several studied functional groups are present in liquid samples, by their IR absorption spectra, was solved with the help of the MLP with nonlinear activation function and error backpropagation algorithm.

Many authors use Raman spectra of compounds for their classification/clusterization using ANN [6, 62, 64, 77]. Thus, the authors [64] showed the possibility of recognition of cellular subtypes, such as the leukocyte subpopulations of lymphocytes and monocytes by spectra of confocal Raman spectroscopy. In this study, three distinct multivariate classification approaches (principal component analysis/linear discriminant analysis (PCA-LDA), support vector machines (SVMs), and random forests) are developed to classify the cellular subtype in extracted peripheral blood mononuclear cells. All methods provided quite satisfactory results. The authors [6] reported the development of an ANN for correct classifying up to 16 different powdery substances: nitrocellulose, TNT, tetryl, picric acid, sodium azide, and others. These studies can be used for the elaboration of compact devices for application in the field of internal security. Using the method of principal component analysis, the dimensionality of input space of spectra of all chemical compounds (in the range 550–1600 cm^{-1}) was reduced, and then a hierarchical cluster analysis was carried out.

For identification of chemical compounds by Raman spectra, the authors [62] proposed to use the convolutional ANN which was trained to automatically identify substances according to their Raman spectra without the need for preprocessing. Raman spectra of mineral compounds from the database RRUFF were used: the detection of three different minerals, selected from a database of 512 spectra of minerals, was carried out simultaneously. A comparative analysis of the large set of methods of subtraction of the pedestal

from the spectra with the case when no subtraction was carried out, and to the original spectra with immediately applied convolutional ANN. It turned out that the convolutional ANN provided greater precision in the application without any preprocessing of spectra compared to any of the investigated methods of preprocessing of the spectra.

ANN methods are used not only to solve the problems of the determination of the type and concentration of certain substances but also to simulate the spectra. In [79], ANN allowed the authors to successfully construct fluorescence spectra of fluorescent materials by their absorption spectra. In order to solve this problem, absorption spectra of a set of fluorophores in the visible region of the spectrum were used as input data. The use of direct-propagation ANN and cascade ANN consisting of two and three hidden layers, combined with the method of PCA, made it possible to predict the fluorescence spectra of the samples with high accuracy (correlation 0.97).

A special place in economics and biomedicine is occupied by the problems of research and control of the quality of various drinks, determining the salt composition of multicomponent aqueous solutions, including industrial and wastewater. In order to solve these problems, various spectroscopic methods are widely used: IR absorption and Raman spectroscopies, fluorescence spectroscopy, and optical absorption spectroscopy. In [69], a method of determination of the content of glucose, fructose, sucrose, and maltose in honey by Raman spectra using PCA, PLS, and ANN was developed. Spectra from 40 model mixtures and 90 honey samples were obtained. The results showed that the methods of PLS and ANN provide close errors in determining the content of these sugars in honey.

Methods of control of alcoholic beverages by their spectra with the involvement of ANN [1, 2, 65, 66] are well developed. Optical absorption spectra, obtained in a wide spectral range, from 190 to 2500 nm, provide information about the content of ethanol, glucose, tartaric acid, etc., which can be used to solve the problem of determining the composition of wine [66]. In this study, the spectra of model aqueous solutions containing ethanol, glucose, tartaric, malic, acetic, lactic acid, and glycerin (7 components of wine) were used. At first, the most informative regions of the spectrum were chosen using the partial least squares (PLS) method. For determination of the composition of the solution, perceptron with one hidden layer was used.

The results showed that the proposed method can reliably detect ethanol, tartaric, and malic acids in solutions, and determine their concentration.

In [2], an attempt to evaluate the quality of wine by FTIR spectra using ANN was made. As parameters characterizing the quality of wine, pH value, total and volatile acidity (concentration of acetic acid), ethanol, and polyphenolic compounds were used. The FTIR spectra of wine samples in the range 400–4000 cm^{-1}, grape variety, year of production, and type of barrel (variety of oak) were used as input data for ANN training. The output parameters were the concentration of phenolic compounds, the type of barrel used, the concentration of acetic acid, and the pH value. The optimal topology of ANN for the solution of this problem was MLP with one hidden layer. Despite the fact that the number of patterns for ANN training was small (only 20), as a result, it was possible to achieve an acceptable accuracy of determining the total and volatile acidity, pH, and content of alcohol and antioxidants.

The authors [1] use shallow convolutional neural networks (CNN) to solve the problems of classification of spectra of beer samples (FTIR, NIR, and Raman), wine (FTIR), drugs in tablets (NIR and Raman), coffee (FTIR), olive oil (FTIR), juice (FTIR), and meat (FTIR). Several classification algorithms were used for comparison: Partial Least Squares Linear Discriminant Analysis (PLS-LDA), Logistic Regression (LogReg), k-Nearest Neighbor (kNN), and CNN. The results showed that when using spectral data without preprocessing, the CNN method significantly exceeds the PLS-LDA method, in most cases it exceeds the LogReg method (except for the case of Raman spectra of drugs in tablets). Preprocessing of spectra (baseline correction, correction of spectra taking into account the contribution of scattering, spectra smoothing, and scaling) in all cases improved the result, and the CNN method was much less dependent on data preprocessing than PLS-LDA.

The method of fluorescence spectroscopy is also quite informative for the study of beverages. It is especially true when it is necessary to solve the problem of assessing the quality of drinking water. This is due to the fact that one of the key parameters determining the quality of water is the type and concentration of organic substances dissolved in it. A successful attempt at analysis of fluorescence/excitation spectra using various data processing methods, including ANN, was performed in [4]. The goal of this study was

to elaborate the method for monitoring the efficiency of removal of dissolved organic matter during purification of drinking water by fluorescent spectra. In this study, self-organizing Kohonen maps and a three-layer feed-forward type of neural network with error back propagation were successfully used. The results showed the prospects of ANN approaches to the analysis of fluorescent spectra.

The concentration of water pollutants can also be determined by the combination of fluorescence spectroscopy and ANN. In [11], the concentration of harmful impurities — anthracene, phenanthrene, and naphthalene — was determined by ANN (three-layer perceptron architecture) using fluorescence spectra of specially prepared solutions of these substances.

13.3.2 *Determination of concentrations of each component in multicomponent salt solutions by Raman spectra. Problem statement and its features*

One of the examples of a complex IP with strong nonlinearity is the problem of identification and determination of the concentration of each salt/ion in multicomponent aqueous solutions of inorganic salts by Raman spectra.

Determination of concentrations of substances dissolved in water is very important for ecological monitoring and control of mineral and waste waters. It is required to be solved in non-contact express mode with acceptable precision. Raman spectroscopy complies with all these requirements.

The principle possibility of using Raman spectra for diagnostics of the salt composition of aqueous media is caused by the high sensitivity of the quantitative characteristics of the spectral bands to the type and concentration of ions dissolved in water. Many salts with complex ions (sulfides, sulfates, nitrates, phosphates, etc.) have their proper Raman bands in the region 300–2000 cm^{-1}. The position of these bands strictly corresponds to the frequency of vibrations of certain molecular groups of ions, and the intensity of the bands depends on their concentration in water. Thus, Raman spectra of water media in this frequency range can be used to determine which complex ions are present in water and in what amount.

However, salts with "simple" ions (e.g. NaCl, KBr, NaI, $CaCl_2$, or $AlCl_3$) do not have their own Raman bands, but they influence the Raman spectral bands of water itself. As it was established in many studies [7, 9, 14, 74], the position and shape of water Raman valence band (2700–4000 cm^{-1}) depends on the types of ions and their concentration in the solution significantly. Thus, the changes in the spectral characteristics of the proper bands of complex ions and of the valence band of water allow simultaneous determination of the types and of concentrations of ions in multicomponent aqueous solutions.

This problem is further complicated by the fact that the studied object is so complex that today it is impossible to build an adequate model for solving the direct problem. Use of neural network methods makes it possible to construct a solution of the desired problem on the basis of experimental spectroscopic data and to increase the accuracy of such a solution in comparison with conventional spectroscopic methods. We solved the problem of identification and determination of concentration of each salt/ion in multicomponent aqueous solutions by Raman spectra with the help of ANN within the framework of two approaches — "experiment-based" and "quasi-model".

13.3.3 *Results of various approaches to the solution of the problem*

To apply the experiment-based approach, we obtained an array of 5147 experimental Raman spectra of aqueous solutions of the following salts: $MgSO_4$, $Mg(NO_3)_2$, LiCl, $LiNO_3$, NH_4F, $(NH_4)_2SO_4$, $KHCO_3$, KF, $NaHCO_3$, and NaCl. The choice of the type and concentration of salts was based on the information about the ions most common in mineral waters and technological water media and their content there.

The prepared solutions of these salts contained from 0 to 10 ions in various combinations and with different concentrations at a constant temperature (24°C) in the range of total ion concentrations from 0 to 1.5 M with increment 0.25 M. As a result, the following sets of spectra were obtained: water — 60 spectra, single-component solutions — 60, two-component — 639, three-component — 1944, five-component — 1376, and six-component — 1128 spectra. Figure 13.1 presents the Raman spectra of the initial single-component aqueous solutions of

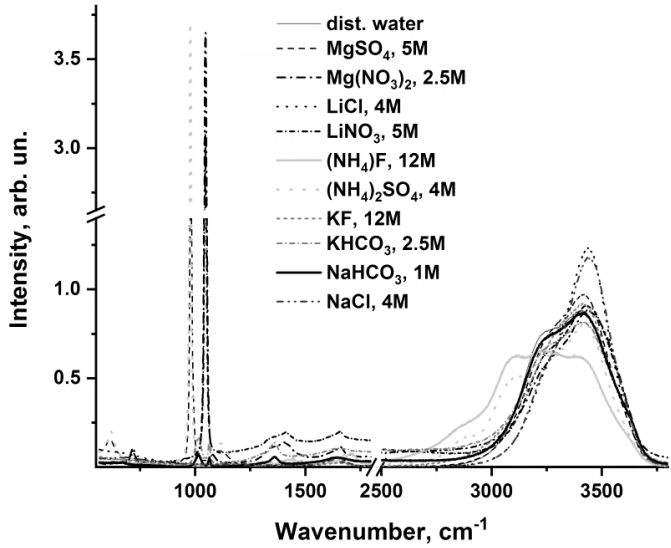


Figure 13.1. Raman spectra of the solutions of all salts with maximum concentration corresponding to their solubility.

all the selected salts with a maximum concentration corresponding to their solubility in water at 20°C. Figure 13.2 presents as examples the obtained experimental Raman spectra of five- and six-component salt solutions with various ion combinations and their content.

For future reference, we call the described problem "10 ions IP", first studied in [22]. However, we started with two similar, but simpler problems: "5 salts IP" [8] and "5 salts clustering" [20]. Use of only 5 salts instead of 10 results in the fact that each cation can be present in the solution only in pair with its specific anion forming the corresponding salt. Relative concentration of this cation and this anion is fixed, allowing one to speak of determination of concentration of salt in spite of the fact that salts in the studied solutions are fully dissociated. The influence of this cation and this anion on the shape of Raman spectrum of the solution is always manifested together, making the IP much less complicated.

The main results that we obtained solving this problem within the experiment-based approach were the following:

(1) Expectedly, the error of the determination of concentrations for the five salts IP was lower when both spectral regions displayed

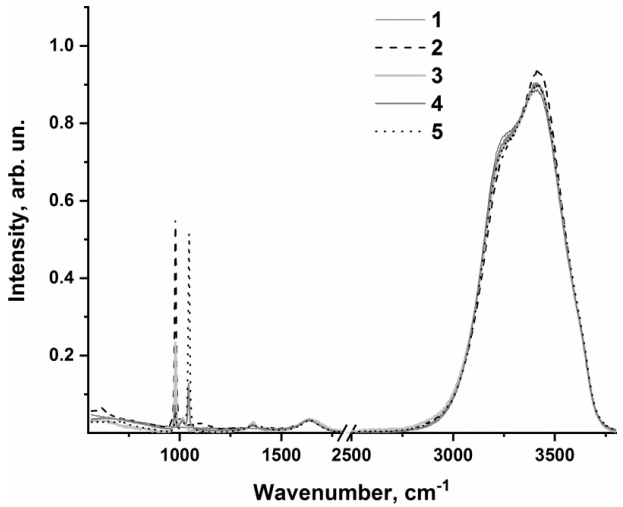


Figure 13.2. Raman spectra of five- and six-component aqueous solutions of salts. 1 — distilled water; 2 — $MgSO_4$ (0.25 M), LiCl (0.25 M), $(NH_4)_2SO_4$ (0.25 M), $KHCO_3$ (0.25 M), NaCl (0.25 M); 3 — NH_4F (0.15 M), $(NH_4)_2SO_4$ (0.2 M), KF (0.15 M), $NaHCO_3$ (0.45 M), NaCl (0.2 M); 4 — $Mg(NO_3)_2$ (0.05 M), LiCl (0.05 M), $(NH_4)_2SO_4$ (0.05 M), $KHCO_3$ (0.2 M), $NaHCO_3$ (0.05 M), NaCl (0.2 M); 5 — $Mg(NO_3)_2$ (0.2 M), LiCl (0.05 M), $LiNO_3$ (0.05 M), $(NH_4)_2SO_4$ (0.05 M), $NaHCO_3$ (0.2 M), NaCl (0.05 M).

in the figures above were used, compared to the case when only the water Raman valence band was used [8, 17].

(2) Input data dimensionality reduction reduces the error of IP solution for all three types of problems [20, 28, 32, 40]. The best improvement was obtained with feature extraction by adjacent channel aggregation for five salts IP (16–18% error reduction on average).

(3) Comparison of the 5 salts and 10 ions IP [29, 30] showed that channel aggregation and autonomous determination of the parameters were efficient only for the simpler problem of 5 salts. The more complicated 10 ions problem requires the use of all the available information both as input and output variables.

(4) There are two basically different approaches to this problem: (a) in a single stage — determination of concentration of each component by the amplitude at the output of the single ANN corresponding to this component and (b) in two stages — first, raw determination of the component composition of the object

by classification or by clustering and second, determination of the concentrations by a separate ANN within each class/cluster [20, 22, 27, 31]. The 2-stage approach turns out to be efficient only for the simpler 5 salts' problems; for the more complex 10 ions' problem it performs worse, most probably due to the small number of patterns in each class/cluster.

It should be noted that the quasi-model approach [8,17] failed to improve the results obtained by the experiment-based approach.

13.3.4 *Application of the elaborated methods to other IP in spectroscopy*

Besides the IP described in detail above, the authors have an extensive experience solving other high-dimension IP in spectroscopy.

Simultaneous determination of temperature and salinity of natural waters [18] required dimensionality reduction for the input data.

A series of studies on ANN identification and determination of concentrations of fluorescing carbon biomarkers [21,58,75,76] allowed us to set grounds for monitoring of excretion of nanodiamonds and nanocomposites used as theranostic agents for drug delivery in human body.

Use of Raman spectroscopy and ANN allowed us to determine concentrations of separate DNA bases in the solution to improve the reliability of molecular DNA computing [23, 24].

Use of the same approach for the diagnostics of water-ethanol solutions allowed us to determine concentrations of alcohol and of harmful impurities in alcoholic beverages [54]. The studies in this direction are being continued.

13.4 Inverse Problem of Geophysics

13.4.1 *Geophysical inverse problem*

The coefficient inverse problems in geophysics consist of finding the coefficients of the systems of differential equations from the measured geophysical fields that are determined by these systems of equations [55]. The coefficients of the systems of the differential equations

are some functions of the physical characteristics of the medium: electrical resistivity, density, magnetization, etc.

The coefficient inverse problems are typically non-linear and in the general case refer to the ill-posed ones. The most efficient and widely applied regularization method in the geophysical inverse problems consists of preliminarily parameterizing the studied media based on the existing *a priori or hypothetical information* and searching for the solution in the finite-parameter classes of media. The parameters of the real media vary within finite limits which makes it possible to reduce the problem of finding the approximate solution of the inverse problem to solving a finite-dimensional operator equation of the first kind on a bounded closed set $\hat{X}_N$ of the finite-dimensional space R^N in the form [55, 82, 83]:

$$F(X) = Y, \quad X \in \hat{X}_N \subset R^N, \quad Y \in \hat{Y}, \tag{13.2}$$

$$X \in \hat{X}_N : \quad x_n \in [x_{n,\min},\ x_{n,\max}], \quad n = 1, \ldots, N, \tag{13.3}$$

where F is a given continuous operator of the direct problem acting the space R^N of the unknown elements to the space $\hat{Y}$ of the measured parameters, $x_{n,\min}$, $x_{n,\max}$ are the minimal and maximal values of the nth parameter, and $Y \in \hat{Y}$ is the element of space of the input data which is determined by the set $Y = (y_1, \ldots, y_M)$ of the measured characteristics of the geophysical field. If the uniqueness conditions of the considered geophysical inverse problem are satisfied on a pair R^N, $\hat{Y}$ of the spaces, the problem of solving Eq. (13.2) with a continuous one-to-one operator F is conditionally well-posed [55, 87] on the compact set $\hat{X}_N$ in form (13.3).

13.4.1.1 *The estimates of the degree of ambiguity of the inverse problem approximate solutions*

For the inverse problems on a compact set, a scalar *a posteriori* error estimate of the found approximate solution can be obtained for a fixed right-hand side y [39]. Let δ be the error level of the approximate right-hand side Y of Eq. (13.1), X_T be the exact solution for the exact right-hand side $Y_T = F(X_T)$, and X_δ be the approximate solution with the residual $\leq \delta : \|F(X_\delta) - Y\|_{\hat{Y}} \leq \delta$. In this case, the following

inequality is true:

$$\|X_\delta - X_T\|_{R^N} \le \beta_1(\delta), \tag{13.4}$$

where $\|\cdot\|_{R^N}$, $\|\cdot\|_{\hat{U}}$ are the given norms in the spaces R^N, $\hat{Y}$, and function $\beta_1(\delta)$ is determined from the solution of the extremum problem:

$$\beta_1(\delta) = \max_{X \in \hat{X}_N} \|X - X_T\|_{R^N}, \quad \|F(X) - Y\|_{\hat{Y}} \le \delta; \quad Y, X_T - \text{fixed}. \tag{13.5}$$

In practice, the error level of the data in the problems of geophysics is typically unknown but the actual residual δ_1 of the obtained approximate solution X_{δ_1}: $\|F(X_{\delta_1}) - Y\|_{\hat{Y}} \le \delta_1$ is known. In this case, the value of function $\beta_1(\delta_1)$ is the scalar characteristic of the degree of ambiguity of the approximate solution X_{δ_1}. It is also possible to obtain local and vector characteristics of the degree of ambiguity [82].

13.4.2 *The approximation artificial neural network (AANN) method for solving the inverse problems*

13.4.2.1 *Reducing the inverse problem to the approximation–interpolation problem*

Irrespective of the inversion method applied, the approximate solutions $\tilde{X} = (\tilde{x}_1, \ldots, \tilde{x}_M)$ of the operator equation (13.2) are the values of a certain vector function specified in the form of the algorithm, namely, function $\tilde{\mathrm{F}}^{-1}(y_1, \ldots, y_M)$ of M variables $y_1, \ldots, y_M$ which are the values of the measured input data that determine the right-hand side of Eq. (13.2). The idea of the approximation approach to solving Eq. (13.2) is to build function $\tilde{\mathrm{F}}^{-1}(y_1, \ldots, y_M)$ in the explicit form using a certain approximation construction [80–83].

13.4.2.2 *Constructing the NN inversion approximator for the inverse problem of geoelectrics*

The ANN method uses neural network constructions. For building an NN function $\tilde{\mathrm{F}}^{-1}(y_1, \ldots, y_M)$, a certain set $\{X_q, Y_q\}$,

$Y_q = F(X_q)$, $q = 1, \ldots Q$ of the known solutions of the direct (and, therefore, inverse) problems is constructed. The construction of the inverse approximating NN function $X = F_{apr}^{-1}(Y)$ is reduced to the problem of training the NN on the $\{X_q, Y_q\}$ set. This problem is solved based on the "model-based" approach (Section 13.2.3, item 2) with the use of various modifications of the error back-propagation algorithm [41].

The high dimensionality of the inverse problems of geoelectrics necessitates solving the problem of compressing the input and output data when training a neural network. The possibility in principle to use compressed data for solving the inverse problem of geoelectrics is a problem of partial inversion based on the incomplete data with the use of local NN approximators of the inversion. This problem was considered in [81] and numerically implemented in [16, 83]. The data compression issue is associated with the problem of selecting the optimal structure of the neural network. The best results for the problems under consideration were obtained with perceptrons with three hidden layers, where the number of neurons in the hidden layers was $L_k = 32, 16, 8;\ k = 1, 2, 3$. The dimension Q of the training set which can reach tens of thousands of solutions of direct problems in practice is another important factor. Calculations have shown that the interpolation properties of the NN approximators that are acceptable for the practice of geoelectrics can be reached at $Q = 10000$ for 2D problems and $Q = 20000$–25000 for 3D problems.

The values of the constructed inverse NN function $F_{apr}^{-1}(Y_q)$ for the input data Y_q from the set $\{X_q, Y_q\}$ approximately coincide with the values of the exact inverse function $X_q \approx F_{apr}^{-1}(Y_q)$ of Eq. (13.2); for the other data Y, the values of the NN function $F_{apr}^{-1}(Y)$ determine the approximate solutions of Eq. (13.2) obtained by interpolating between the elements of the training set $\{X_q, Y_q\}$. Thus, function F_{apr}^{-1} is an approximate inverse operator of Eq. (13.2) and is often referred to as the NN *inversion approximator*. If necessary, the approximate solution obtained by the NN approximator can be refined by the approximation-iterative neural network (AINN) method based on constructing the correcting NN approximators [82, 83]. The correcting NN approximators are built on the training set of the reference solutions lying in the vicinity of the solution found at the first iteration.

13.4.2.3 *Intrinsic interpolation error of the NN inversion approximator*

The approximate solution $X_{apr} = F_{apr}^{-1}(Y)$ of the inverse problem (13.2) obtained by the inversion approximator F_{apr}^{-1} is by its sense interpolative. The error of the solution X_{apr} due to the inaccuracy of the inverse operator differs from zero even in the case of the precise (accurate) data. The deviation of $X_{apr} = F_{apr}^{-1}(Y_T)$ from the true solution X_T of the inverse problem (13.2) does not exceed value ε_{apr} [81–83]:

$$\|X_{apr} - X_T\|_{R^N} \leq \varepsilon_{apr}, \tag{13.6}$$

where ε_{apr} is the intrinsic interpolation error of the trained approximator F_{apr}^{-1}, which is determined from the solution of the extremum problem:

$$\varepsilon_{apr} = \max_{X \in \hat{X}_N} \|F_{apr}^{-1}(F(X)) - X\|_{R^N}. \tag{13.7}$$

Alongside the general interpolation error ε_{apr} of the approximator over the entire set of the parameters, it is also possible to calculate the interpolation error ε_{apr}^{i} for a certain selected ith group of the parameters, e.g., for the parameters of the ith tier of the parameterization grid.

13.4.2.4 *Well-posedness issues in the problem of learning the NN inversion approximator*

The problem of training the NN inversion approximator F_{apr}^{-1} is ill-posed because of the absence of uniqueness of its solution [83]. The purpose of the training task is to construct an inversion approximator with the minimum interpolation error of the form (13.7) that is acceptable for practice. Therefore, from the set of the solutions of the extremum problem, the one is selected at which the intrinsic interpolation error of the inversion approximator is minimal. This selection is implemented by calculating the values of the approximator on the patterns of the validation set using the technique of "restarting" and "viewing" the set of the extrema of the optimization training process.

Function of the NN approximator $F_{apr}^{-1}(Y)$ is continuous for any $Y \in \hat{Y}$, which ensures the stability of the approximate solutions of the inverse problem.

13.4.2.5 *The inversion of the geoelectric field data with the use of the NN inversion approximator*

For the obtained approximate solution $X_\delta = F_{apr}^{-1}(Y)$, the residual $\bar{\delta} = \|F(X_\delta) - Y\|_{\hat{Y}}$ and the *a posteriori* local and vector characteristics of the degree of ambiguity in accordance with Section 13.4.1.1 are calculated. The time of data inversion using the trained approximator is determined by the speed of calculating the function of the neural network and is typically a few seconds. The time for calculating the residual and the ambiguity characteristics is determined by the speed of solving the direct problem.

13.4.2.6 *Application of the NN methods in the inverse problems of geophysics*

The applications of the NN methods for solving the inverse problems of geophysics are reviewed in [81–83]. The general concept of applying the NN technologies in geophysics is described in [73]. Neural network methods are currently actively used in seismic prospecting and in other geophysical methods for data processing, constructing various types of dependencies, clustering, etc. However, the application of these methods for solving inverse problems in the classical sense, i.e., for solving the operator equations of the first kind of type (13.2) mainly is limited to the problems of geoelectrics. The examples of solving the 1D inverse geoelectric problem with the use of neural networks are presented in [42, 72]. In [80, 84], it was shown that based on the neural networks it is possible to obtain approximate solutions of low-parametric 2D and 3D inverse geoelectrical problems where the number of the sought parameters is about $N \sim 10 \div 15$. By adapting the NN methods to the peculiarities of the problem to be solved, we managed to generalize the NN method to the case of the 2D and 3D media whose geoelectric properties are determined by $N \sim n \cdot 10^2 \div n \cdot 10^3$ parameters [81–83].

13.4.3 *Examples of solving the geophysics inverse problems by NN method*

The efficiency of the NN technology in geophysics is illustrated by the examples of solving the inverse problem of geoelectric (magnetotelluric (MT) method) using synthetic and field data.

13.4.3.1 *The example solution of the inverse 3D MT problem based on the synthetic data*

For the 3D MT inversion, we constructed the inversion approximator $\Pi_1^K(3D)$ which does not need specifying the first approximation and solves the inverse 3D (MT) problem in the class of the grid models for the depths of at most 4.2 km provided that the impedance tensor Z and tipper W of the MT field are specified on the space–frequency grid covering the region of measurements.

3D-K1 Model: Figure 13.3(a) shows the 3D model of the medium for which the synthesized data were calculated. The number of the sought parameters of this model is $N_3 = 532$. Figure 13.3(b) shows the result of the inversion of the exact synthesized data at the first iteration. For better visualization, the images of the block model are presented in the smoothed form obtained by linear interpolation between the blocks.

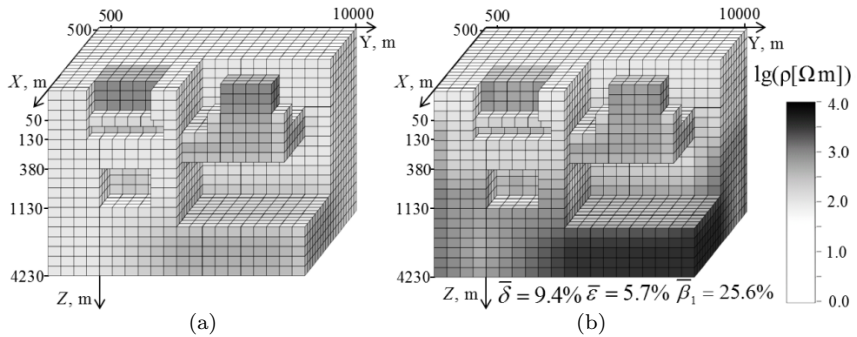


Figure 13.3. The inversion results for the 3D-K1 model: (a) the original 3D-K1 model with the frontal fragments cut out; (b) the results of the inversion with the use of the NN approximator $\Pi_1^K(3D)$ at the first iteration; $\bar{\delta}$ is the weighted average residual of the solution, $\bar{\varepsilon}$ is the mean error of the solution over all the model parameters, and $\bar{\beta}_1(\bar{\delta})$ is the estimate of the degree of ambiguity of the inverse problem solution.

The values shown in Fig. 13.3 are the mean error $\bar{\varepsilon}$ of the inversion over the model parameters; weighted average residual $\bar{\delta}$ of the obtained solution, and the estimate of the degree of ambiguity $\bar{\beta}_1(\bar{\delta})$ of the solution of the inverse problem.

13.4.3.2 *The example solution of the inverse 2D MT problem based on the field data*

As an example of the NN inversion of the field data, we present the segment of the regional profile 2DV from 1530 to 2100 km [34]. Figure 13.4 illustrates the results of the NN inversion of these data after their office preprocessing. As a result of solving the inverse problem on a regularized parameterization grid, 1292 parameters were determined within the limits of the considered 2D region shown in Fig. 13.4. The first approximation was not specified. The residuals of the solution calculated for the individual segments of the profile are shown in Fig. 13.4. The estimates of the degree of ambiguity of

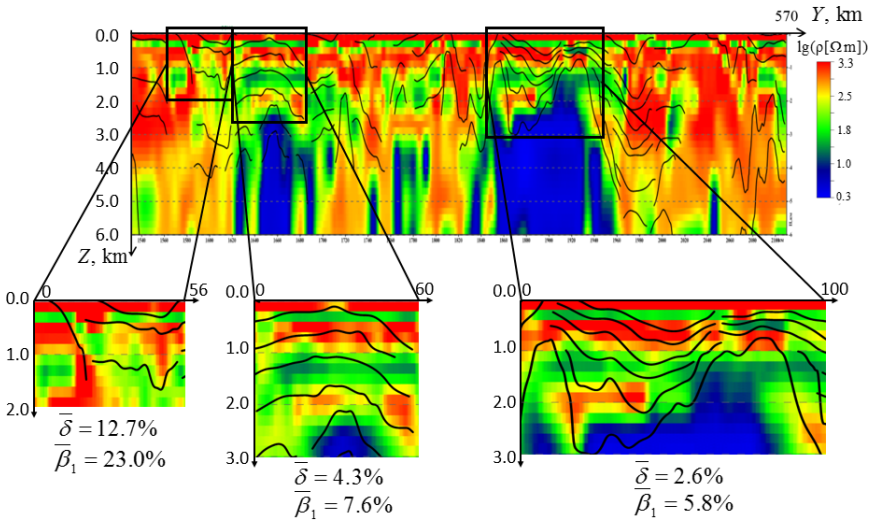


Figure 13.4. Geoelectric section along the regional profile 2DV, segment 1 (1530–2100 km): (a) NN inversion, 1292 parameters determined. Black rectangles mark the individual areas for the detailed analysis. Black curves are the boundaries of the geological layers according to the seismic data. $\bar{\delta}$ is the weighted average residual of the solution and $\bar{\beta}_1(\bar{\delta})$ is the estimate of the degree of ambiguity of the inverse problem solution.

the inverse problem solution for these segments vary from 2 to 12%. Overall, the residual over the entire segment of the profile is 11.7% and the degree of ambiguity of the inverse problem solution (average over all the tiers of the grid) is 20.4%.

For comparison, the resulting geoelectrical section (Fig. 13.4) is superimposed by the independent seismic data (black curves).

13.5 Conclusion

Neural network methods of solving inverse problems and methodological aspects of their application were considered in this chapter. Theoretical consideration and a number of examples of use prove the fact that properly used artificial neural networks are a powerful instrument for the solution of high-dimensional inverse problems.

Acknowledgments

This study has been performed at the expense of the grant of Russian Science Foundation, project No 19-11-00333 https://rscf.ru/en/project/19-11-00333/.

References

[1] Acquarelli, J., van Laarhoven, T., *et al.* (2017) Convolutional neural networks for vibrational spectroscopic data analysis. *Analytica Chimica Acta*, 954, 22–31.

[2] Agatonovic-Kustrin, S., Morton, D.W., and Yusof, A.P.M. (2013) The use of Fourier transform infrared (FTIR) spectroscopy and artificial neural networks (ANNs) to assess wine quality. *Modern Chemistry & Application*, 1(4), 1000110.

[3] An, G. (1996) The effects of adding noise during backpropagation training on a generalization performance. *Neural Computing*, 8(3), 643–674.

[4] Bieroza, M., Baker, A., and Bridgeman, J. (2011) Classification and calibration of organic matter fluorescence data with multiway analysis methods and artificial neural networks: an operational tool for improved drinking water treatment. *Environmetrics*, 22, 256–270.

[5] Bishop, C.M. (1995) Training with noise is equivalent to Tikhonov regularization. *Neural Computing*, 7(1), 108–116.

[6] Botti, S., Dipoppa, G., *et al.* (2014) Raman spectra massive classification using artificial neural networks. *ENEA*, 31 p. www.enea.it.
[7] Burikov, S.A., Dolenko, T.A., *et al.* (2005) The effect of hydration of ions of inorganic salts on the shape of the Raman stretching band of water. *Optics and Spectroscopy*, 98(2), 235–239.
[8] Burikov, S.A., Dolenko, S.A., *et al.* (2010) Application of artificial neural networks to solve problems of identification and determination of concentration of salts in multi-component water solutions by Raman spectra. *Optical Memory and Neural Networks*, 19(2), 140–148.
[9] Busing, W.R. and Hornig, D.F. (1961) The effect of dissolved KBr, KOH or HCl on the Raman spectrum of water. *Journal of Physical Chemistry*, 65, 284–292.
[10] Cancilla, J.C., Díaz-Rodríguez, P., *et al.* (2014) Artificial neural networks applied to fluorescence studies for accurate determination of N-butylpyridinium chloride concentration in aqueous solution. *Sensors and Actuators B*, 198, 173–179.
[11] Cauchi, M., Bianco, L., and Bessant, C. (2011) The quantification of pollutants in drinking water by use of artificial neural networks. *Natural Computing*, 10, 77–90.
[12] Cirovic, D. (1997) Feed-forward artificial neural networks: Applications to spectroscopy. *Trends in Analytical Chemistry*, 16(3), 148–155.
[13] Dolenko, S.A., Dolenko, T.A., *et al.* (1999) Solution of inverse problem in nonlinear laser fluorimetry of organic compounds with the use of artificial neural networks. *Pattern Recognition and Image Analysis*, 9(3), 510–515.
[14] Dolenko, T.A., Churina, I.V., *et al.* (2000) Valence band of liquid water raman scattering: Some peculiarities and applications in the diagnostics of water media. *Journal of Raman Spectroscopy*, 31, 863–870.
[15] Dolenko, T.A., Fadeev, V.V., *et al.* (2002) Fluorescence diagnostics of oil pollution in coastal marine waters using artificial neural networks. *Applied Optics*, 41(24), 5155–5166.
[16] Dolenko, S., Guzhva, A., *et al.* (2009) Comparison of adaptive algorithms for significant feature selection in neural network based solution of the inverse problem of electrical prospecting. Lecture Notes in Computer Science, Vol. 5769, pp. 397–405.
[17] Dolenko, S.A., Burikov, S.A., *et al.* (2012a) Adaptive methods for solving inverse problems in laser Raman spectroscopy of multicomponent solutions. *Pattern Recognition and Image Analysis*, 22(4), 551–558.
[18] Dolenko, S., Dolenko, T., *et al.* (2012b) Comparison of input data compression methods in neural network solution of inverse problem

in laser Raman spectroscopy of natural waters. Lecture Notes in Computer Science, Vol. 7553, pp. 443–450.

[19] Dolenko, S., Isaev, I., *et al.* (2013) Study of influence of parameter grouping on the error of neural network solution of the inverse problem of electrical prospecting. Communications in Computer and Information Science (CCIS), Vol. 383, pp. 81–90.

[20] Dolenko, S., Burikov, S., *et al.* (2014a) Neural network approaches to solution of the inverse problem of identification and determination of partial concentrations of salts in multi-component water solutions. Lecture Notes in Computer Science, Vol. 8681, pp. 805–812.

[21] Dolenko, T.A., Burikov, S.A., *et al.* (2014b) Optical imaging of fluorescent carbon biomarkers using artificial neural networks. *Journal of Biomedical Optics*, 19(11), 117007.

[22] Dolenko, S., Efitorov, A., *et al.* (2015) Neural network approaches to solution of the inverse problem of identification and determination of the ionic composition of multi-component water solutions. Communications in Computer and Information Science (CCIS), Vol. 517, pp. 109–118.

[23] Dolenko, T.A., Burikov, S.A., *et al.* (2016) Adaptive methods of solving inverse problems for improvement of fidelity of molecular DNA computations. *Optical Memory and Neural Networks*, 25(1), 16–24.

[24] Dolenko, T.A., Burikov, S.A., *et al.* (2017) Improvement of reliability of molecular DNA computing: Solution of inverse problem of Raman spectroscopy using artificial neural networks. *Laser Physics*, 27(2), 025203.

[25] Dougherty, G. (2013) Pattern Recognition and Classification: An Introduction. (Springer-Verlag, New York).

[26] Dramićanin, T., Dimitrijević, B., and Dramićanin, M.D. (2011) Application of supervised self-organising maps in breast cancer diagnosis by total synchronous luminescence spectroscopy. *Applied Spectroscopy*, 65, 293–297.

[27] Efitorov, A.O., Burikov, S.A., *et al.* (2015a) Comparison of the quality of solving the inverse problems of spectroscopy of multi-component solutions with neural network methods and with the method of projection to latent structures. *Optical Memory and Neural Networks*, 24(2), 93–101.

[28] Efitorov, A., Burikov, S., *et al.* (2015b) Significant feature selection in neural network solution of an inverse problem in spectroscopy. *Procedia Computer Science*, 66, 93–102.

[29] Efitorov, A., Dolenko, T., *et al.* (2016a) Neural network solution of an inverse problem in Raman spectroscopy of multi-component solutions of inorganic salts. Advances in Intelligent Systems and Computing, Vol. 449, pp. 273–279.

[30] Efitorov, A., Dolenko, T., *et al.* (2016b) Solution of an inverse problem in Raman spectroscopy of multi-component solutions of inorganic salts by artificial neural networks. Lecture Notes in Computer Science, Vol. 9887, pp. 355–362.

[31] Efitorov, A., Dolenko, T., *et al.* (2018a) Study of efficiency of dividing the problem space as a means to improve solution of multi-parameter inverse problem by adaptive methods. *Procedia Computer Science*, 123, 122–127.

[32] Efitorov, A., Dolenko, S., *et al.* (2018b) Use of adaptive methods to solve the inverse problem of determination of composition of multi-component solutions. *Optical Memory and Neural Networks*, 27(2), 89–99.

[33] Fadeev, V.V., Dolenko, S.A., *et al.* (1997) Laser diagnostics of complicated organic compounds and complexes by saturation fluorimetry. *Quantum Electronics*, 27(6), 556–559.

[34] Fel'dman, I.S., Okulesskii, B.A., *et al.* (2008) Electrical exploration by MTS method in a complex of regional oil and gas prospecting works in the European part of Russia. *Journal of Mining Institute*, 176, 125–131 (In Russian).

[35] Fessendenand, R.J. and Gyorgyi, L. (1991) Identifying functional groups in IR spectra using an artificial neural network. *Journal of the Chemical Society, Perkin Transactions*, 2, 1755–1762.

[36] Fletcher, R. (1987) *Practical Methods of Optimization*, 2nd Ed. (Wiley, USA).

[37] Gajjar, K., Trevisan, J., *et al.* (2013) Fourier-transform infrared spectroscopy coupled with a classification machine for the analysis of blood plasma or serum: A novel diagnostic approach for ovarian cancer. *Analyst*, 138, 3917–3926.

[38] Gerdova, I.V., Churina, I.V., *et al.* (2002) New opportunities in solution of laser spectroscopy inverse problems using artificial neural networks. *Izvestiya Akademii Nauk Serie Physics*, 66(8), 1116–1124 (in Russian).

[39] Goncharskii, A.V. and Yagola, A.G. (1969) On the uniform approximation of the monotonic solution of ill-posed problems. *Doklady Akademii Nauk SSSR*, 184(4), 771–773 (in Russian).

[40] Gushchin, K.A., Burikov, S.A., *et al.* (2015) Data dimensionality reduction and evaluation of clusterization quality in the problems of analysis of composition of multi-component solutions. *Optical Memory and Neural Networks*, 24(3), 186–192.

[41] Haykin, S. (2009) *Neural Networks and Learning Machines*, 3rd Ed. (Pearson Education), 937 p.

[42] Hidalgo, H., Gómez-Treviño, E., and Swiniarski, R. (1994) Neural network approximation of a inverse functional. *IEEE International Conference on Neural Networks*, Vol. 5, pp. 3387–3392.

[43] Holmstrom, L. and Koistinen, P. (1992) Using additive noise in back-propagation training. *IEEE Transactions on Neural Networks*, 3(1), 24–38.

[44] Isaev, I.V. and Dolenko, S.A. (2014) Improving the accuracy of neural network solution of the inverse problem of electrical prospecting by sequential determination of parameters: Verification on model data. *Proceedings of 10th International Conference "Problems of Geocosmos"*, October 6–10, 2014, St. Petersburgh, Russia, pp. 17–22. http://geo.phys.spbu.ru/materials_of_a_conference_2014/C2014/02_Isaev.pdf.

[45] Isaev, I.V. and Dolenko, S.A. (2016a) Comparative analysis of residual minimization and artificial neural networks as methods of solving inverse problems: Test on model data. Advances in Intelligent Systems and Computing, Vol. 449, pp. 289–295.

[46] Isaev, I.V. and Dolenko, S.A. (2016b) Training with noise as a method to increase noise resilience of neural network solution of inverse problems. *Optical Memory and Neural Networks*, 25(3), 142–148.

[47] Isaev, I. and Dolenko, S. (2018a) Adding noise during training as a method to increase resilience of neural network solution of inverse problems: Test on the data of magnetotelluric sounding problem. Studies in Computational Intelligence, Vol. 736, pp. 9–16.

[48] Isaev, I. and Dolenko, S. (2018b) Training with noise addition in neural network solution of inverse problems: Procedures for selection of the optimal network. *Procedia Computer Science*, 123, 171–176.

[49] Isaev, I. and Dolenko, S. (2019) Group determination of parameters and training with noise addition: Joint application to improve the resilience of the neural network solution of a model inverse problem to noise in data. Advances in Intelligent Systems and Computing, Vol. 848, pp. 138–144.

[50] Isaev, I., Obornev, E., *et al.* (2016) Increase of the resistance to noise in data for neural network solution of the inverse problem of magnetotellurics with group determination of parameters. Lecture Notes in Computer Science, Vol. 9886, pp. 502–509.

[51] Isaev, I.V., Burikov, S.A., *et al.* (2018a) Improving the resilience of neural network solution of inverse problems in Raman spectroscopy of multi-component solutions of inorganic compounds to the distortions caused by frequency shift of the spectral channels. *Journal of Physics: Conference Series*, 1096(1), 012100.

[52] Isaev, I., Burikov, S., *et al.* (2018b) Joint application of group determination of parameters and of training with noise addition to improve the resilience of the neural network solution of the inverse problem in spectroscopy to noise in data. *Lecture Notes in Computer Science*, 11139, 435–444.

[53] Isaev, I., Vervald, E., *et al.* (2018c) Neural network solution of an inverse problem in Raman spectroscopy of multi-component solutions of inorganic salts: Group determination as a method to increase noise resilience of the solution. *Procedia Computer Science*, 123, 177–182.

[54] Isaev, I., Burikov, S., *et al.* (2019) Artificial neural networks for diagnostics of water-ethanol solutions by Raman spectra. Studies in Computational Intelligence, Vol. 799, pp. 167–175.

[55] Kabanikhin, S.I. (2008) Definitions and examples of inverse and ill-posed problems. *Journal of Inverse and Ill-Posed Problems*, 16, 317–357.

[56] Kuzniz, T., Tur, M., *et al.* (2000) Neural network analysis of absorption spectra and optical fiber instrumentation for the monitoring of toxic pollutants in aqueous solutions. *Proceedings of SPIE*, Vol. 4185, pp. 444–447.

[57] Kuzniz, T., Halot, D., *et al.* (2007) Instrumentation for the monitoring of toxic pollutants in water resources by means of neural network analysis of absorption and fluorescence spectra. *Sensors and Actuators B: Chemical*, 121, 231–237.

[58] Laptinskiy, K., Burikov, S., *et al.* (2016) Monitoring of nanodiamonds in human urine using artificial neural networks. *Physica Status Solidi A: Applications and Materials Science*, 213(10), 2614–2622.

[59] Lavrentiev, M.M., Romanov, V.G., and Shishatskii, S.P. (1986) *Ill-Posed Problems of Mathematical Physics and Analysis* (American Mathematical Society, Providence).

[60] Lechowicz, Ł., Urbaniak, M., *et al.* (2013) The use of infrared spectroscopy and artificial neural networks for detection of uropathogenic *scherichia coli* strains' susceptibility to cephalothin. *Acta Biochimica Polonica*, 60(4), 713–718.

[61] Lenhardt, L., Zeković, I., *et al.* (2013) Artificial neural networks for processing fluorescence spectroscopy data in skin cancer diagnostics. *Physica Scripta*, T157, 014057–014060.

[62] Liu, J., Osadchy, M., *et al.* (2017) Deep convolutional neural networks for Raman spectrum recognition: A unified solution. *Analyst*, 142, 4067–4074.

[63] Liu, W., Yang, W., *et al.* (2008) Use of artificial neural networks in near-infrared spectroscopy calibrations for predicting glucose concentration in urine. Lecture Notes in Computer Science, Vol. 5226, pp. 1040–1046.

[64] Maguire, A., Vega-Carrascal, I., *et al.* (2015) Competitive evaluation of data mining algorithms for use in classification of leukocyte subtypes with Raman microspectroscopy. *Analyst*, 140(7), 2473–2481.

[65] Malek, S., Melgani, F., and Bazi, Y. (2018) One-dimensional convolutional neural networks for spectroscopic signal regression. *Journal of Chemometrics*, 32, e2977.

[66] Martelo-Vidaland, M.J. and Vázquez, M. (2015) Application of artificial neural networks coupled to UV–VIS–NIR spectroscopy for the rapid quantification of wine compounds in aqueous mixtures. *CyTA — Journal of Food*, 13(1), 32–39.

[67] Matsuoka, K. (1992) Noise injection into inputs in back-propagation learning. *IEEE Transactions on Systems, Man, and Cybernetics*, 22(3), 436–440.

[68] Mouwen, D.J.M., Capita, R., *et al.* (2006) Artificial neural network based identification of *Campylobacter* species by Fourier transform infrared spectroscopy. *Journal of Microbiological Methods*, 67, 131–140.

[69] Ozbalci, B., HakkıBoyacı, I., *et al.* (2013) Rapid analysis of sugars in honey by processing Raman spectrum using chemometric methods and artificial neural networks. *Food Chemistry*, 136, 1444–1452.

[70] Piotrowski, A.P. and Napiorkowski, J.J. (2013) A comparison of methods to avoid overfitting in neural networks training in the case of catchment runoff modeling. *Journal of Hydrology*, 476, 97–111.

[71] Pizzi, N., Choo, L.-P., *et al.* (1995) Neural network classification of infrared spectra of control and Alzheimer's diseased tissue. *Artificial Intelligence in Medicine*, 7, 67–79.

[72] Poulton, M., Sternberg, B., and Glass, C. (1992) Neural network pattern recognition of subsurface EM images. *Journal of Applied Geophysics*, 29(1), 21–36.

[73] Raiche, A. (1991) A pattern recognition approach to geophysical inversion using neural nets. *International Journal of Geophysics*, 105, 629–648.

[74] Rull, F. and De Saja, J.A. (1986) Effect of electrolyte concentration on the Raman spectra of water in aqueous solutions. *Journal of Raman Spectroscopy*, 17(2), 167–172.

[75] Sarmanova, O.E., Burikov, S.A., *et al.* (2018a) Optical visualization and control of the excretion of theranostic fluorescent nanocomposites from the body using artificial neural networks. *Nanomedicine, Nanotechnology, Biology, and Medicine*, 14, 1371–1380.

[76] Sarmanova, O.E., Burikov, S.A., *et al.* (2018b) Monitoring of the excretion of fluorescent nanocomposites out of the body using artificial neural networks. *Proceedings of SPIE*, Vol. 10679, p. 1067904.

[77] Sattlecker, M., Bessant, C., *et al.* (2010) Investigation of support vector machines and Raman spectroscopy for lymph node diagnostics. *Analyst*, 135, 895–901.
[78] Schmitt, J., Udelhoven, T., *et al.* (1998) Artificial neural networks applied to FTIR and FT-Raman spectra in biomedical applications. *Proceedings of AIP Conference*, Vol. 430, p. 260.
[79] Shams-Nateri, A. and Piri, N. (2015) Predicting emission spectra of fluorescent materials from their absorbance spectra using the artificial neural network. *Optica Applicata*, 45(4), 545–557.
[80] Shimelevitch, M.I. and Obornev, E.A. (1998) The method of neuron network in MTS inverse problems. *Proceedings of 14th Workshop on Electromagnetic Induction in the Earth*, p. 159 (In Russian).
[81] Shimelevich, M.I. and Obornev, E.A. (2009) An approximation method for solving the inverse MTS problem with the use of neural networks. *Izvestiya, Physics of the Solid Earth*, 45(12), 1055–1071.
[82] Shimelevich, M.I., Obornev, E.A., *et al.* (2017) The neural network approximation method for solving multidimensional nonlinear inverse problems of geophysics. *Izvestiya, Physics of the Solid Earth*, 53(4), 588–597.
[83] Shimelevich, M. I., Obornev, E. A., *et al.* (2018) An algorithm for solving inverse geoelectrics problems based on the neural network approximation. *Numerical Analysis and Applications*, 11(4), 359–371.
[84] Spichak, V.V. and Popova, I.V. (1998) Application of the neural network approach to the reconstruction of three-dimensional geoelectric structure. *Izvestiya, Physics of the Solid Earth*, 34(1), 33–39.
[85] Tanabe, K., Tamura, T., and Uesaka, H. (1992) Neural network system for the identification of infrared spectra. *Applied Spectroscopy*, 46(5), 807–810.
[86] Terekhov, S.A. (1997) Direct, inverse and combined problems in complex engineered system modeling by artificial neural networks. *Proceedings of SPIE AeroSense Conference*, Orlando, Florida, 21–24 April 1997 (*Proceedings of SPIE*, Vol. 3077, Paper 71).
[87] Tikhonov, A.N. and Arsenin, V.Y. (1977) *Solutions of Ill-Posed Problems.* (Wiley, New York).
[88] Trevisan, J., Angelov, P.P., *et al.* (2012) Extracting biological information with computational analysis of Fourier-transform infrared (FTIR) biospectroscopy datasets: current practices to future perspectives. *Analyst*, 137, 3202–3215.
[89] Vapnik, V. (1982) Estimation of Dependencies Based on Empirical Data (Springer-Verlag, New York).
[90] Wang, C. and Principe, J.C. (1999) Training neural networks with additive noise in the desired signal. *IEEE Transactions on Neural Networks*, 10(6), 1511–1517.

[91] Wang, Q., Zheng, N., Li, Z., and Ma, Z. (2015) Quantitative analysis of glucose in whole blood using FT-Raman spectroscopy and artificial neural network. *Proceedings of International Conference on Computational Science and Engineering, ICCSE*, pp. 471–475.

[92] Wenning, M., Büchl, N.R., and Scherer, S. (2010) Species and strain identification of lactic acid bacteria using FTIR spectroscopy and artificial neural networks. *Journal of Biophotonics*, 3, 493–505.

[93] Wu, W., Walcmk, B., *et al.* (1996) Artificial neural networks in classification of NIR spectral data: Design of the training set. *Chemometrics and Intelligent Laboratory Systems*, 33, 35–46.

[94] Yin, S., Liu, C., *et al.* (2015) Noisy training for deep neural networks in speech recognition. *EURASIP Journal on Audio, Speech, and Music Processing*, 2015(2), 1–14.

[95] Zarnowiec, P., Mizera, A., *et al.* (2016) Chemometric analysis of attenuated total reflectance infrared spectra of Proteus mirabilis strains with defined structures of LPS. *Innate Immunity*, 22(5), 325–335.

[96] Zur, R.M., Jiang, Y., *et al.* (2009) Noise injection for training artificial neural networks: A comparison with weight decay and early stopping. *Medical Physics*, 36(10), 4810–4818.

This page intentionally left blank

© 2025 World Scientific Publishing Company
https://doi.org/10.1142/9789811267215_0014

Chapter 14

Generalized Spectral-Analytical Method and Its Applications in Problems of Image Analysis and Pattern Recognition

Sergey A. Makhortykh*, Liudmila I. Kulikova†, Anton N. Pankratov‡, and Ruslan K. Tetuev§

Russian Academy of Sciences, Keldysh Institute of Applied Mathematics, Institute of Mathematical Problems of Biology, Pushchino, Moscow Region, the Russian Federation

**makh@impb.ru*

†*likulikova@mail.ru*

‡*pan@impb.ru*

§*ruslan.tetuev@gmail.com*

The generalized spectral-analytical method is based on the use of orthogonal decompositions of signals and analytic transformations in the space of expansion coefficients. The theory of classical orthogonal bases is a generalization of the theory of Fourier series to algebraic orthogonal polynomials of continuous and discrete arguments. The choice of the optimal set of features is carried out on the basis of integral signal estimates, and the mathematical operations on the signals studied are performed in the space of the expansion coefficients. Within the framework of the generalized spectral-analytical method, the problems of adaptive analytical description of one-dimensional and multidimensional digital information arrays are effectively solved in order to reduce the redundancy of signals' description, as well as recognition problems

on the basis of pairwise comparison of objects in Fourier coefficients features space.

This chapter describes the application of this method in the following problems:

- image recognition (by using invariants when recognizing contour objects),
- bioinformatics (recognition of repeats in sequences and complex spatial configurations of biomacromolecules),
- biomedicine (analysis of the spatial and temporal organization of electrical and magnetic activity of the brain).

Keywords: Orthogonal bases, Fourier series, magnetic encephalography, the generalized spectral-analytical method, the Karunen–Loeve basis, the Haar functions, Walsh functions, Chebyshev polynomials, Kravchuk polynomials, Charlier polynomials, Meixner polynomials, Hahn polynomials, Jacobi polynomials, Legendre polynomials, Hermite polynomials, Laguerre polynomials, Gibbs phenomenon, Fourier bases and wavelets, inverse problem, spherical harmonics, source localization, audible stimulus, Kotelnikov theorem, Frenet formula, dynamic programming.

14.1 Introduction

Tendencies to an increase in the volume of information arrays and an increase in the requirements for accuracy and completeness of their processing in the shortest time force the developers of new information systems to increase their computing power. A continuous increase in the speed of computing systems leads to an increase in the cost of processing information, by no means always ensuring that the required processing conditions are met. Thus, at present, the task of creating new data processing methods, which along with ensuring a given accuracy and processing speed, would not necessarily require an increase in the power of computing systems, remains an urgent task.

The proposed paper describes the results of a search for possible ways of combining digital calculations with analytical conversions and derivations in order to improve the accuracy and speed of calculations on a computer. At the same time, the visibility of the analytical outputs and the possibility at each step of the calculations to have an analytical presentation of the calculated characteristics, the desired parameters, and estimates are preserved.

The successful implementation of the combined data processing method directly depends on the form of the analytical description of the original numeric arrays. An analysis of possible methods for obtaining an analytical description showed [1, 12–16, 25, 30] that the most fully established conditions are met by a method based on approximation of data by segments of orthogonal series using classical orthogonal polynomials and functions of continuous and discrete arguments [16]. Remarkable approximative properties of orthogonal bases determine their attractiveness when solving the previously mentioned problems, and the use of approximation results in various analytical transformations and outputs to obtain the required estimates or characteristics make classical orthogonal bases a promising tool in the existing set of methods and approaches for analyzing digital information arrays.

The theory of classical orthogonal bases is a continuation of the theory of Fourier series to algebraic polynomials. Their distinctive feature consists mainly in the fact that in the majority of formulas defining concrete bases, there are parameters whose change can noticeably modify the properties of orthogonal polynomials and weight functions forming a particular orthogonal basis. The latter circumstance is especially important in problems of optimal analytical approximation, when a given accuracy must be ensured by the shortest segment of the orthogonal series. The use of special adaptive procedures ensures optimal (quasi-optimal) analytical approximation of data in automatic mode.

Along with the theoretical foundations, this chapter presents examples of the application of **the generalized spectral-analytical method** (**GSAM**) for solving a number of practical problems. The task of localizing the sources of biomagnetic activity of the brain in magnetic encephalography (MEG) data records, image analysis and recognition using contours approximation, and analysis of genomes and protein molecules. In this case, the features of the object of recognition are the coefficients of the expansion of the function under study on the most appropriate orthonormal basis.

The developed algorithms provide a solution to the following tasks: an analytical description of the data, compression of the amount of information provided, filtering low-frequency and high-frequency noise, implementation of the signal recovery procedure in the areas of lack of records, implementation of procedures for the

normalization and scaling of the signal, determine the coordinates and orientation of records relative to each other, and implementation of the spectral recognition of signals. A further increase in the efficiency of this method is associated with solving the problem of accelerating the procedure for calculating the coefficients of the decomposition.

The efficiency of decomposition on one or another non-adaptive basis depends on the signal type (class) $x(t)$, which is naturally a significant inconvenience for the user if we don't have a priori information about the process being analyzed. Here, efficiency is the minimization of the mean-square error of the expansion for a given number of expansion terms, or vice versa, the minimization of the number of expansion terms for a given error [14, 16]. The Karunen–Loeve basis [29, 68] is considered to be the best in this respect, based on the statistical characteristics of $x(t)$. However, the difficulties encountered in the practical implementation of the spectral decomposition in this optimal basis, level its merits.

In practice, the use of the apparatus of classical orthogonal bases to solve problems of the analytical description of information arrays is due to their remarkable approximate capabilities. Important properties of classical orthogonal bases, which determine their importance in the problems of data analytical approximation are discussed in detail in [12–14, 16, 45].

In a generalized and systematic form, these classical polynomials are described by Nikiforov and Uvarov [45]. They considered a differential equation, particular solutions of which are the classical orthogonal polynomials, as well as special functions of mathematical physics: spherical, cylindrical, and hypergeometric.

Use of the orthogonal bases of a discrete variable is caused by the presence of tasks in which the process under study is determined by known values at discrete points in time or when the analyzed signal allows its discrete representation.

In such problems, of course, one can use the orthogonal bases of a continuous argument. In this case, when recovering a signal, its value should be determined only at discrete points in time. However, the calculation of the coefficients of decomposition of the basis of a continuous argument requires the implementation of time-consuming numerical integration operations. The use of orthogonal bases of piecewise constant functions (such as the Haar and Walsh functions)

in such problems is also impractical since discrete-variable functions are most natural for computer processing and free from the drawbacks of continuous and piecewise constant functions.

Orthogonal bases of a discrete variable distinguishing feature is that they all have from one to three variable parameters, the change of which within certain limits significantly changes the properties of one or another orthogonal basis, as well as the nature of the weight functions, called in a discrete version, jump functions.

It is clear that these bases are divided into two classes. The first class includes orthogonal Chebyshev and Kravchuk systems with a fixed interval of existence. The bases of Charlier, Meixner, and Hahn constitute another class — they are determined on a semi-infinite interval.

Interesting area of application for GSAM is computer vision, more specifically, image recognition based on contour analysis. The silhouettes of the objects are the most informative part of the visible world, and it is believed that our primitive ancestors had no color vision, and the shapes were the only source of visual information [7]. Still the raster-based analysis and similarity tests would never be a yes-no question because of the noise effects — the same object shapes would be similar but never identical in every pixel. Thus, we came to a brightness-based simple idea: let's estimate similarity for raster images simply as a count of the pixels where the images match [5].

Many applications are not suitable for brightness-based matching when we allow more complex deformations of contours like bending. Some researchers started considering silhouettes as contours, which means they are just two-dimensional closed lines, so thus we get naturally sorted boundary points, which makes it much easier to compare a pair of rows instead of inner pixels. For example, if there are square-shaped objects to be compared and their sizes are 100×100, following the brightness-based approach, we have to compare 10,000 pairs of pixels, while it takes only 400 points when a contour-based method is used and that is 25 times less data to compute. We can speed up the matching even more than that using data decimation [2]: if we leave only every 10th point of a boundary line, we get simply 40 points, and thus we increase the speed even 10 times more. In doing this we must understand that we no longer compare the silhouettes of the objects but we compare some polygons with 40 vertices and hope

that those polygons are sufficiently good in describing the original shapes.

The decimated boundary curve for decades become a *de facto* standard of representation for the contour-based approach, but here we use another way to compress boundary data and show how to estimate shape similarity with no need to decompress the curves back. The ability to do calculations in the frequency domain is one of the most important advantages of GSAM; we call it *feasibility*, which arose from a simple fact: Fourier coefficients are fully competent representations of an original function and completely capable of performing any task about the functions. In [59], were demonstrated what formula to implement and what computation technique to use in a spectral analysis if we need a derivative of spectrally represented functions, or integral, or if we should solve a nonlinear equation entirely in a frequency domain. The work mostly focused on the classical orthogonal polynomials (Jacobi, Legendre, etc.), and the polynomials of a discrete variable (Hahn, etc.), and the whole family [46].

Why use so many different polynomials? At the very beginning of the GSAM development, there was a question: Which one of these orthogonal polynomials is the best? Later, it was clearly demonstrated that different applications of spectral analysis bring better results using different bases. We call that feature *adaptiveness*; this means we may consider our method in general, not only for a single basis like DCT, and that is why there is a 'G' in GSAM. For instance, when dealing with bell-shaped signals, the Hermite polynomials might be the best choice for a better approximation, but we use Laguerre ones for fading oscillating signals. Thus, we significantly shorten the approximation and reduce artifacts as the Gibbs phenomenon.

At the moment, many algorithms and programs [3,34,37,48] have been developed for the computational assessment of the similarity of DNA fragments and its derivatives (proteins and RNA). However, over the past 20 years, the possibilities of computational analysis of genetic data have lagged behind the rapidly developing experimental base in modern biology: most algorithms for processing genetic sequences are based on several basic principles of processing textual information, such as Hamming or Levenstein edit distances. The time

complexity of algorithms is essentially nonlinear, the main slowing factor when comparing similar genetic data is point mutations (such as substitutions, deletions, and insertions of letters), the correction of which increases the analysis time. For larger sizes of compared fragments, it is reasonable to assume a greater number of mutations and, as a result, to observe a sharp decline in the efficiency of algorithms in large areas (10,000 bp long).

The proposal to use the spectral approach to the problem of searching for repeats in the genomes was first made in [18, 19] in which ideas were put forward to use the methods of continuous mathematics to analyze symbol sequences. As a functional counterpart of genomic sequence, it was proposed to use the curves of GC-content, which characterize the strength of the binding of the DNA double helix.

The next stage in the development of the method was the construction of a dot matrix of similarity on the base of a decision rule for the recognition of inexact repeats [50]. To search for extended tandem repeats, an estimate of the periodicity of the GC-content curve was proposed [61]. As an application of the developed methods, extended repeats were discovered [62] in the genomes that set the task of automating the search for repetitive sequences of this type.

In the course of further research, the method has received a number of improvements [51]: the repeats recognition stability is increased by introducing an additional GA-content curve, which makes the informational description of the sequence complete; an improved decision rule, invariant to the choice of scale, is derived; the approximation conditions were determined, under which tandem repeats are displayed on a dot matrix in the form of square patterns, which allowed to automate the recognition of repeats by a sample, and to find out a new repeat in the genome [55]; an algorithm for searching inverted repeats in the space of decomposition coefficients is constructed, and the first results on genome-wide comparison are presented.

In the course of further studies proposed are theoretical basis and algorithmic implementation of the spectral-analytical method of recognition of repeats in character sequences. The theoretical grounds are based on the theorem on equivalent representation of the

character sequence by the vector of continuous characteristic functions [52, 53]. Comparison of fragments of characteristic functions is performed in the standard metric in Euclidean space of expansion coefficients of the Fourier series of orthogonal polynomials. An essential feature of this approach is the ability to evaluate repeats at different scales. Another important feature is the possibility of efficient parallelization of data. In the development of algorithms, we preferred a scheme of computing with a minimal amount of references to memory, implying repetitive calculations and evaluations on demand. In this paradigm, proposed is an algorithm for calculating the coefficients of expansions in the orthogonal polynomials through the use of recurrence relations. It is shown that the algorithm for calculating the coefficients of expansions in the orthogonal polynomials can be effectively vectorized by computing with a fixed vector length. Parallelization and vectorization were implemented using the OpenMP standard extension of language C/C++. The developed method effectively scales, depending on the parameters of the problem and the number of processor cores on systems with shared memory.

SBARS software was developed [56] to search for inexact repeats of various types (direct, inverted, or tandem) in genomes and compared with other tools [21, 32, 36, 58]. In addition, a universal web service for global alignment of extended sequences was built [63].

At present, attention of researchers is drawn to structural motives having unique spatial arrangements [23]. This interest is caused by the uniqueness of their structures and their capacity to serve as nuclei in the course of protein folding. However, irrespective of the mechanism by which the protein folding proceeds, structural motives can be used as starting structures in the course of identification of possible arrangements of a polypeptide chain in modeling the protein structure [23, 24, 31].

In earlier works [23, 31], some simple rules of a polypeptide chain arrangement were revealed. As a result, structural trees describing a diversity of protein structures were constructed. This made possible automatic recognition and prediction of various structural motives of protein molecules.

The simplest structural motives are structural motifs consisting of two elements of the secondary structure: α-α-corners, α-α-hairpins, L-shaped and V-shaped structures, and others [8, 23, 24, 31].

14.2 Principal Statements of GSAM

Spectral decomposition is any transformation process using orthogonal basis functions $\{P_i(t)\}$ in a series

$$x(t) = \sum_{i=1}^{N} a_i P_i(t), \tag{14.1}$$

where a_i are the decomposition coefficients and $x(t)$ is the stationary process from L_2, i.e.

$$\int_{-\infty}^{\infty} |x(t)|^2 < \infty. \tag{14.2}$$

The system of functions $\{P_i(t)\}$ is considered orthogonal on the interval $t_1 < t < t_2$ if

$$\int_{t_1}^{t_2} P_i(t) \cdot P_j(t)dt = \delta_{ij} = \begin{cases} 0 & \text{if } i \neq j, \\ 1 & \text{if } i = j. \end{cases} \tag{14.3}$$

If $\delta_{jj} = 1$, the system $\{G_i(t)\}$ is orthonormal. The system of functions is considered as quasi-orthonormal if the condition is met:

$$\int_{t_1}^{t_2} P_i(t) \cdot P_j(t)dt = d_{ij} = \begin{cases} \xi & \text{if } i \neq j, \\ 1 & \text{if } i = j, \end{cases}$$

where $\xi > 0$.

Any system of n linearly independent functions $\{P_i(t)\}$ can be transformed into an orthonormal system of functions using Gram–Schmidt operations. Spectral decomposition can be carried out based on the use of classical orthogonal polynomials $\{P_i(t)\}$ (Chebyshev, Legendre, Hermite, etc.) [13, 45], trigonometric functions (Fourier methods), discontinuous, or sequential functions [38].

As a result of the decomposition procedure (14.1), the transition is made from a point-wise signal recording to its spectral representation. All further analysis and processing operations are performed over the spectrum. Proposed technique is adaptive to the class of signals. It consists of a preliminary assessment of data type and on its results a choice of the optimal basis is performed. The correctness of the choice provides an approximation of maximal simplicity in this

class of data. Approximation tuning to the desired function system is performed by the signal's vector **K** of "aspect ratios" [14, 16]. Its components are the set of signal features. The problem of spectral expansion optimization is reduced, in fact, to the problem of optimal basis recognition in the feature space of coefficients:

$$K_i = \frac{(g_i, f)}{(h_i, f)}. \tag{14.4}$$

(*,*) is the scalar product or projection of function f on the template functions g_i, h_i with mutually contrasting properties. They may be, for example, the first basis functions of estimated basic systems. The use of adaptive procedures increases the efficiency of analytical description and provides a high rate of data volume compression.

After the conversion from the original recordings to the coefficients of expansion, all operations on the signals are carried in the space of spectral features. They include pattern recognition, data classification, and diagnosis, as well as the more common mathematical operations: integration, differentiation, multiplication, division, raising to integer and non-integer degree, optimal filtering, calculation of statistical moments, dimensional characteristics, and so on. For these purposes, the special library of mathematical transformations has been developed for different bases (including those for functional systems of several variables). Some aspects of this library will be mentioned later. To obtain the needed characteristics of the original record, we have a standard method of orthogonal series segment calculation. This finite orthogonal series is the required approximation of the signal characteristics. The procedures of computational technology are efficiently realized on parallel computers so that we have practically linear scaling of both spectrum calculation and subsequent operations on the spectrum. Some basic procedures can be implemented in hardware in the form of special expansion computing devices for mainframes.

14.3 Vectorization and Analytical Description

The generalized spectral-analytic method is based on the following statements:

(1) *Parametric description (vectorization)*: The object under study is described by a system of parametric equations of one argument:

$$\{x_i = X_i(t); i = 1, \ldots, n. \tag{14.5}$$

For example, a signal representing the variation of some quantity in time is described by a single equation ($n = 1$) and a plane curve (in particular, the image contour) is unambiguously described by a system of two equations:

$$x = X(t), \quad y = Y(t). \tag{14.6}$$

(2) *Approximation*: Parametric equations (projections) are approximated by truncated orthogonal series with a preset accuracy (in the uniform or root-mean-square sense) as it follows from (14.1):

$$X(t) = \sum_{k=0}^{N_x} A_k P_k(t), \quad Y(t) = \sum_{k=0}^{N_y} B_k P_k(t), \tag{14.7}$$

where the expansion coefficients are given by the integrals

$$A_k = \int_a^b X(t) P_k(t) 1 p(t) dt, \quad B_k = \int_a^b Y(t) P_k(t) p(t) dt, \tag{14.8}$$

(3) and $p(t)$ is the weight function of the selected base $\{P_k\}$.
In order to attain the needed accuracy with the minimal length of series (2.3), we need to select an appropriate system $\{P_k\}$. Such series constitute an analytical description of the original object. The required accuracy of description and the number of coefficients vary according to the problem at hand.

(4) Further processing hinges on the fact that the expansion coefficients carry all essential information about the object. This means that having found the expansion coefficients, we can calculate all the intrinsic characteristics of the object or use the coefficients as features for the recognition problem. The dimension of the feature space can be further reduced through the use of the statistical properties of expansion coefficients for a given class of images.

The developed data processing technology based on these statements is supported by the following considerations:

- the parametric description of multidimensional objects (for example, plane and space curves) reduces the processing of multidimensional data to a set of one-dimensional problems,
- the adaptive analytical description of each projection by the most suitable orthogonal polynomials results in a drastic reduction of the data-representation volume (called data compression),
- further data processing uses analytical manipulations that can be performed in advance, thus eliminating completely the need for multiple processing of numerical arrays,
- the suggested procedures ensure full processing of data in the compressed form without restoring original objects.

To illustrate the technology proposed, we can apply it to the analysis of the plane curve [8]. The use of parametric representation of such an object requires the accomplishment of certain additional procedures, namely:

- outlining the contour image of the object under study,
- vectorizing the image and rendering its projections to a form convenient for analytical description,
- numbering the contour image and determining the initial point of numbering,
- specifying the parameters of the contour image in a convenient coordinate system.

Contour outlining and vectorization are independent problems and are presently described in detail. In some problems of data processing (for example, in computer cartography), images are represented as sequences of pairs (x_i, y_i), which constitutes the desired parametrization. Special programs are used for the vectorization of raster images.

For closed contours, it is difficult to provide a standard choice of the reference point on the contour from which a parameter is reckoned. This problem can be solved by optimizing a certain functional, depending on the reference point.

Any plane curve can be described analytically with a desired accuracy by a system of equations such as (14.7).

The calculation of integrals (14.8) is the most critical and laborious operation in the spectral analysis. It is here that the change to known functions of one variable is effected, allowing to take full advantage of analytical manipulations. Gauss quadrature formulas optimal for given types of polynomials are used for the calculation of integrals (14.8):

$$A_k = \int_{-1}^{1} X(t)P_k(t)dt \approx \sum_{j=1}^{n} \lambda_j X(t_j)P_k(t_j), \qquad (14.9)$$

where the values of $X(t_j)$ at the quadrature nodes are found by interpolation. This recipe tangibly reduces the amount of calculations (by the order of magnitude for a 256×256 raster) without losing essential details and permits to obtain rather long approximation series if necessary ($N_x, N_y \sim 1000$).

Turning to the use of expansion coefficients as features, we obtain a space of dimension $N_x + N_y$, instead of the initial dimension $M \times M$, where M is the raster size. Another advantage of this approach is the weak dependence of the number of calculations on the raster size in the image, which is due to the fact that the number of points in the contour increases linearly with M.

Equations (14.9) describe the shapes of the plane configurations to be studied. The expansion length varies according to the required accuracy of contour description. Thus, fewer terms of the orthogonal series are needed to get a general idea of the object's shape, whereas a greater depth of expansion is necessary to take into account minute details.

The orthogonal functions in equations (14.7) may not be from the same family, but, in a number of cases, it is useful that they should belong to the same base.

14.4 Optimization of the Analytic Description

It is very important to choose the most suitable orthogonal base for the approximation of curves. As a rule, it is necessary to strive for the optimal analytical approximation of each curve under analysis. There are two steps of the analytic description optimization:

(5) The choice of an appropriate coordinate system. For example, a circle is described by trigonometric functions, which require

N_x, N_y ∼10–15 for approximation in the Cartesian coordinate system and in polar coordinates, it is accurately described by $N_\rho = 0, N_\varphi = 1$.

(6) The choice of the proper base and the base parameters. It is evident that for good approximation the resemblance between the shape of the examined curve projection and the first orthogonal functions of the chosen basis is significant. Automation of the analytic representation of the initial data with the most suitable orthogonal base requires introducing of a procedure which would select for every signal such a base out of the set of bases entered into the computer.

In order to estimate the input signal, the concept of a "form coefficient" was introduced (according to (14.4)):

$$K_f = \int_a^b X(t)\eta(t)dt \cdot \left[\int_a^b X(t)\xi(t)dt\right]^{-1}, \tag{14.10}$$

where $X(t)$ is the function (projection) under study and $\eta(t), \xi(t)$, are known functions. As follows from the definition, (14.10) is a ratio of the scalar products of the examined signal with the known functions.

Computing the integrals, we obtain the value characterizing the respective contributions of the contrasting functions.

Let us consider a simple case $\eta(t) = T - t$ and $\xi(t) = t$ on the interval $[0, T]$, then we obtain

$$K_f = \int_0^T X(t)(T-t)dt \cdot \left[\int_0^T X(t)tdt\right]^{-1}. \tag{14.11}$$

The integral in the numerator of (14.11) estimates the magnitude of the linearly decaying component in $X(t)$, and the integral in the denominator of (14.11) assesses the increasing component. The ratio (14.11) gives the following results: if $K > 1$, then $X(t)$ decays on $[0, T]$, if $K < 1$, then $X(t)$ increases on $[0, T]$, and if $K \sim 1$, then $X(t)$ is periodical or quasi-constant.

To organize the optimal approximation, we must first estimate the properties of the orthogonal bases by the first few weighted polynomials assumed by the form coefficient. During the processing of $X(t)$, the computer estimates its form and performs the approximation with the basis, the form coefficient of which is nearest to the form coefficient found for the signal in question.

14.5 GSAM Applications

14.5.1 *Magnetoencephalography data*

Magnetoencephalography (MEG) is a recently developed experimental technique [20,39–42] for studying human neural activity, mapping functional brain areas, and diagnosing various disorders. Research in this area is motivated by the possibility of noninvasive imaging of cortical and subcortical processes.

Until recently, the technique was difficult to apply because of the extreme weakness of both spontaneous and evoked neuromagnetic fields generated by current sources. Accordingly, the measuring apparatus to be employed must meet particularly stringent requirements.

Despite technical difficulties and the high cost of the required apparatus, this technique can be used as a complementary or alternative to electroencephalography because magnetic field is much less perturbed by intracranial inhomogeneities and integumentary tissues than electric field, and the accuracy of source localization can therefore be substantially improved without detailed knowledge of the intracranial structure.

The input experimental data used in this chapter were obtained by means of a 148-channel Magnes 2500 WH neuromagnetometer (a highly sensitive SQUID-based magnetic field measurement system) at the New York University School of Medicine. The system was housed in a magnetically shielded room.

The analysis presented in this chapter is focused on the recognition and classification of patterns of magnetic field distribution over the subject's scalp for localizing neuromagnetic activity sources. This problem arises from the complex variability of an MEG signal during the measurement time (approximately 10–20 min), which includes alternating intervals of abnormal and normal activity. Intervals of abnormal and normal activity can easily be identified visually when they differ by the mean amplitude of the signal. Otherwise, visual inspection cannot be reliably used to identify intervals corresponding to different activity modes, and the dominant type of activity can be determined by analyzing the spatial field pattern, which strongly correlates with the number, location, and orientation of the current sources responsible for magnetic field generation.

The proposed approach was validated against data derived from a reference group of healthy individuals and a group of patients suffering from Parkinson's syndrome. The signal was a 148-dimensional time series measured at 148 points on the head surface at a sampling rate of 500 Hz per channel.

The data analysis can be divided into the following tasks:

- detection of a signal associated with a particular type of brain activity (e.g. response to auditory, visual, tactile, or other stimulation and generation of Parkinson's tremor or auditory hallucinations),
- selection of instants for sampling the spatial field distribution over the head surface to be used as input in solving the inverse problem of current source localization,
- solution of the inverse problem of source localization in the cases of abnormal and normal activity,
- the use of fMRI data as physiological constraints.

14.5.2 *Frequency analysis*

It can be shown by means of spectral routines that oscillatory components contained in the MEG signal power change regarding the system status. For this purpose, frequency or time-frequency analysis based on Fourier bases and wavelets were used.

As an example, monaural audible stimulus was applied. Pulsing frequency was 7 Hz, while the response was localized at frequencies near 10 and 20 Hz. Figure 14.1 shows this situation on the wavelet diagram [39–41].

14.5.3 *The inverse problem*

The locations and orientations of the sources of spontaneous activity are determined by using the magnetic field values at the measurement points to solve the inverse problem. For the adopted model of brain activity and head geometry, the inverse problem can generally be reformulated as a nonlinear optimization problem for the calculation of the locations and moments of the set of dipoles that provide the best approximation of the measured MEG data with respect to RMS deviation. The inverse problem is difficult to solve primarily because

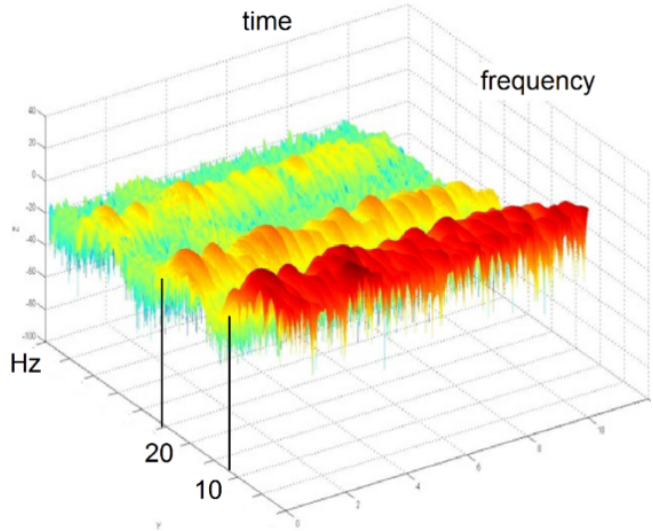


Figure 14.1. Haar wavelet diagram of MEG response for 7 Hz pulsing audible stimulation.

it is ill-posed: there exist infinitely many solutions consistent with the experimental data. The computational complexity of the nonlinear optimization problem is explained by the existence of multiple local extrema of the objective function, particularly when the number of dipole sources is large [66].

There also exist technical difficulties due to the weakness of both spontaneous and evoked magnetic fields generated by brain current sources against a high background noise level in MEG measurements.

To classify the types of neuromagnetic activity and identify abnormal intervals in MEG records, we use the feature space spanned by the spectral expansion coefficients of the total magnetic field on a sphere, which is invariant under rotation. The natural basis for describing the spatial field distribution is the set of spherical harmonics:

$$f(\theta,\phi)=\sum_{n=0}^{N}\sum_{s=0}^{n}(a_{ns}p_n^s(\cos\theta)\cos s\phi+b_{ns}p_n^s(\cos\theta)\sin s\phi).$$

Here p_n^s are the associated Legendre functions and a_{ns} and b_{ns} are expansion coefficients. The Fourier coefficients of expansions of MEG data in spherical harmonics are used as relevant features in

the proposed classification method. The data sample used to train the pattern recognition algorithm is the set of vectors of expansion coefficients for all MEG signals corresponding to different number, location, and orientation of spontaneous activity sources. Note that training by exhaustive search and comparison of measured and training patterns is too labor-consuming because of the high dimension of the training feature space. Relevant features are extracted at the training stage of the algorithm by estimating the variability of the expansion coefficients for the class of objects under analysis. The upper bound for the dimension of the feature space depends on the length N of the Fourier expansion. There exists a simple analytical relationship between the expansion coefficients of functions whose arguments are related by an SO(2) transformation [67], which makes it possible to develop a fast search procedure for functions in a particular class.

14.5.4 *Source analysis*

Source estimation comprises two steps: (1) estimation of the potential or field distribution for a known source and for a known model of the head is referred to as direct modeling; (2) estimation of the unknown sources corresponding to the measured MEG is referred to as inverse modeling.

The head model specifies how currents that are generated by sources in the brain, e.g. dipoles, are influenced by the tissue properties and how these result in externally measurable MEG fields.

To reconstruct the location and the time course or spectral content of a source in the brain, various source localization methods are available. Here calculations with audible stimulation data and Parkinson patients' records used MRIAN software [66]. It is rather efficient for one- and two-dipole problems with the usage of structural restrictions by means of fMRI tomograms.

The distribution of the localized two current dipoles for the normal and pathological signal according to the results of visual analysis and on the basis of spectral pre-processing is shown in Figures 14.2(a) and 14.2(b), respectively.

Results of induced and spontaneous sources' localizations are presented in Figures 14.3 and 14.4. It shows the response to the monaural audible stimulus in the auditory cortex (right brain: 20 Hz in the

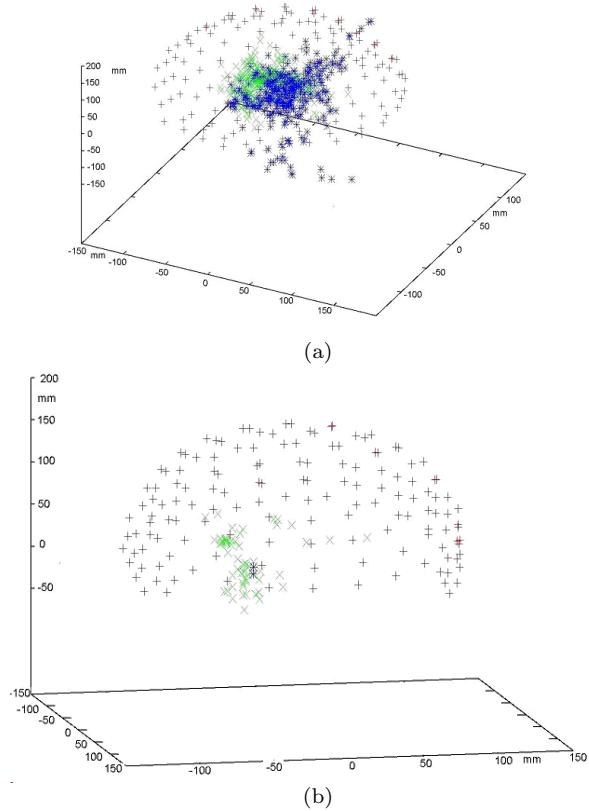


Figure 14.2. The distribution of the localized two current dipoles according to the results of visual analysis (a); spectral pre-processing (b).

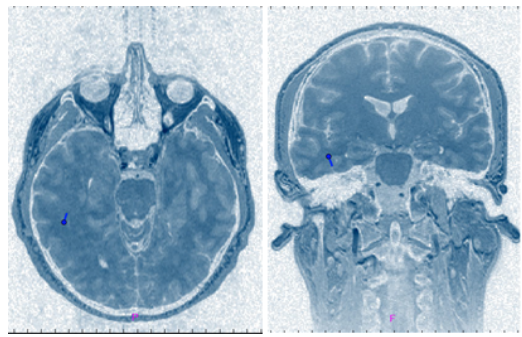
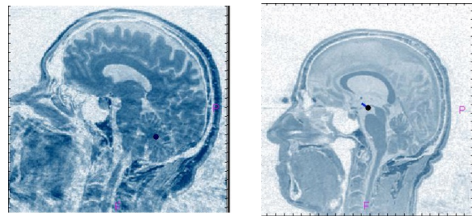


Figure 14.3. Sources' localization (circles with rods) for induced activity (audible stimulus response).



(a) (b)

Figure 14.4. Sources' localization for spontaneous Parkinsonian signal: (a) source in the cerebellum; (b) source in substantia nigra.

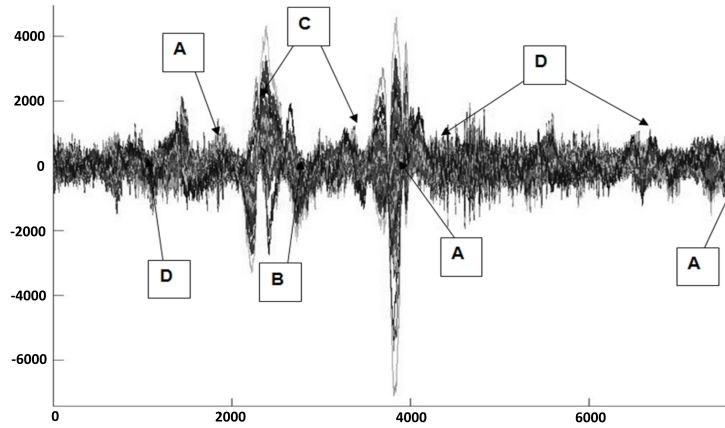


Figure 14.5. Temporal dynamics of activity regimes A, B C, D switches on time series of MEG records.

ipsilateral zone, Figure 14.3). Figures 14.4(a,b) demonstrate localized sources of pathological activity in left brain zones for the case of Parkinsonian patients.

Temporal dynamics of activity regime switches is shown in Figure 14.5. Full description is brought out in [39–41].

14.6 Contour-Based Shape Recognition

Here we present a simpler solution based on a classic Fourier transform of the sine/cosine functions. So, as we mentioned above, we can estimate shape similarity with no need to decompress their boundary curves back. Note that a parametric representation is used to

describe a curve S, which is $S(t_i) = \{X(t_i), Y(t_i)\}$, where $X(t)$, $Y(t)$ are the coordinates of a point at the moment t. This representation is an *absolute path* but a good alternative is a *relative path*, less depending on a coordinate system: $S(t_i) = \{V(t_i), K(t_i)\}$, for instance, described by its velocity $V(t_i)$ and its signed curvature $K(t_i)$, and that representation linearly correlated to geometric transformations as rotations and scaling. Besides, since we do not care how fast the point runs, let's take a velocity as some constant value $V(t_i) = const$, so, by doing this, we reduce the shape data twice. Indeed, we surely can track, say a Formula-1 pilot, if we knew its velocity and steering wheel position at every moment t, now let's assume that the bolide velocity equals 60 Mph all race long — definitely, we still can reconstruct its path, despite such a loss of data.

Assuming that a model shape is defined by curvature function $S = \{K(t_i)\}$, where t_i varying between -2π and 2π while i lays in range $[0, T]$. After FFT, this would be represented in a frequency domain as $(A_0, B_o, A_1, B_1, \ldots, A_N, B_N)$:

$$K(t_i) = \sum_{n=0}^{N} A_n \cos nt_i + B_n \sin nt_i.$$

Let us suppose that the shape we need to compare with a model is of the same size and we were lucky to choose a corresponding starting point for both S and $s = \{k(t_i)\}$ of a spectral row $(a_0, b_o, a_1, b_1, \ldots, a_N, b_N)$. In that case, we can propose a simple estimation for shape similarity as a root mean square:

$$r = \sqrt{\sum\nolimits_{i=0}^{T} (S(t_i) - s(t_i))^2 \Delta t_i / T}, \tag{14.12}$$

which in accordance with Parseval's identity equals to

$$r = \sqrt{\sum\nolimits_{n=0}^{N} ((A_n - a_n)^2 + (B_n - b_n)^2)}. \tag{14.13}$$

The identity of (14.12) and (14.13) fits well with our idea of comparing shapes totally in the frequency domain. We can set a certain threshold value and all the pairs of shapes with r below that threshold should be considered similar.

Then we use simple band-pass filters of a cut-off frequency; this leaves only spectral coefficients with the indices below its value and

truncates the 'tail' of the spectrum, which is another kind of spectral approximation used broadly for its simplicity. Cut-off filters are the exact equivalent of the data decimation according to the Kotelnikov theorem. We can claim that when we are leaving every 10th point of the boundary line, we lose details just as much as if we leave the first 10% of the spectrum. Anyway, whether the similarity is estimated in the time or frequency domain, one should find a compromise solution between opposite cases: the lesser data would spare much of memory space, shorten the runtime, and make a noise reduction, but if there are too few data left, we will get too much false positive errors.

Previously, we supposed for simplicity that the shapes we are comparing would be of the same size and the starting points always be chosen correctly. Actually, the first condition is relatively easy to fulfill; all the shapes could be normalized to have the same size, but choosing the unique starting point for any closed lines such, that it stays stable on every image no matter what kind of distortions it suffers is quite a challenge! There are a lot of solutions for that problem proposed by researchers [11]; still it is hard to propose a universal one, but we choose one when the phase information of the Fourier coefficients is ignored, and only their magnitudes are matched. Despite the fact that we throw away half of the data, this approach works well to be a 'weak detector' of similarity. Still we might use the rest of the data as the magnitude-based similarity tests are passed [59], so, using these double-checks we can propose a 'strong-detector'. Previously, we defined the fixed depth of spectral representation as 20 coefficients, so, in that case for a weak detector, we need only these 10 values $(C_0, C_1, \ldots, C_9)$:

$$C_n = \sqrt{A_n^2 + B_n^2}. \tag{14.14}$$

14.7 Recognition of Repeats in Genomes

To adapt the spectral-analytical approach to problems in bioinformatics needed to generalize the concept of dot matrix. At the same time, this approach can be applied to the search for repeats in any symbolic or functional sequences. In this chapter, the approach is described in the most general form with the preservation of some terminology and examples from the field of bioinformatics.

The method is based on the spectral decomposition of functions that make up the characteristic description of a text sequence, in which the following properties of functions are important: (1) completeness and (2) continuity. The completeness of the description means that the original sequence can be reconstructed from the characteristic curves. The property of continuity is necessary for the evaluation of repeats in the form of changes in characteristics.

These conditions are satisfied by the curves of the content of subsets of nucleotides in a window of a given length along the sequence of the macromolecule. This type of curve includes the well-known GC-content curve that has been studied in bioinformatics. At the same time, the window size, which is a parameter of such a description, actually introduces the concept of scale for a character sequence.

For the general case, the following theorem holds:

Theorem 14.1. *For an arbitrary character sequence in the alphabet of M characters, there are $Log_2 M$ characteristic functions from which the original sequence can be restored, and the functions are smoothly varying K-valued, where K is the scale parameter.*

Proof. Let us encode the symbols of the sequence with a numerical vector in the binary number system. It takes at least $Log_2 M$ bits. Now consider the sliding window of width K, and sum up the number of units of a certain bit of all the characters in the sequence in this window. The function defined in this way, which depends on the beginning of the position of the window in the sequence, is called the characteristic function of the sequence corresponding to the specified bit of the binary character encoding. Each bit of the character in a sequence can be recovered from its corresponding characteristic function.

The characteristic curves that make up the description of an object are broken down into overlapping fragments of length W with shift d. After that, all fragments f_i, g_i considered as discrete functions with numbering of samples within the length W of the window are compared in pairs, based on the standard metric in Euclidean space:

$$\rho(f,g) = (f-g, f-g) = \frac{1}{W}\sum_{i=1}^{W}(f_i - g_i)^2.$$

To reduce calculations of the distances between fragments, approximation of fragments of characteristic functions by segments of an

orthonormal series is used. Therefore, the distance is estimated by the formula:

$$\rho(f,g) = \sum_{i=1}^{N}(c_i - d_i)^2,$$

where c_i, d_i is the coefficients of the Fourier series of N number (and, $N \ll W$). The use of spectral decomposition makes it possible not only to economically estimate the distance but also to perform transformations to estimate inverted and complementary sequences in the space of decomposition coefficients, which means simultaneous recognition of all types of repeats without transforming the sequence itself.

The threshold decision rule is used to recognize repeats: if $\rho < \varepsilon$, where ε is the threshold value, then fragments are considered similar, but if $\rho \geq \varepsilon$, fragments are not similar. If there are several characteristic curves that make up a complete description of an object, they are recognized simultaneously, and the result is a logical multiplication of the decision rules for each of the characteristic functions. This approach improves the robustness of recognition to errors. This follows from the fact that the decision rule is fulfilled in the region of the minimum of the metric ρ, considered as a function of the fragment number. Thus, the set of minima determines the set of candidates for repeats. In the case of two features, for example, GC- and GA-curves, the set of repeats is taken as the intersection of the sets of repeat candidates received for each of the signs separately. After these operations, the results of the comparison are displayed on a dot matrix, one point on which, however, corresponds to the comparison of two whole fragments, and not just the sites of the sequence. The dot matrix is one of the illustrative standard representations of the results of the comparison of two sequences, which makes it possible to display the alignment of inaccurate repeats, as well as their relative position. The generalized dot matrix allows you to get new opportunities to align inaccurate repeats. For example, it was shown that inaccurate extended tandem repeat can be displayed as a perfect square on a dot matrix (Figure 14.6). This is achieved due to the correct selection of the ratio between the size of the window and the shift of the approximation window. On the basis of this important result, a fully automated method for the recognition of tandem repeats was constructed and previously unknown repeats were found.

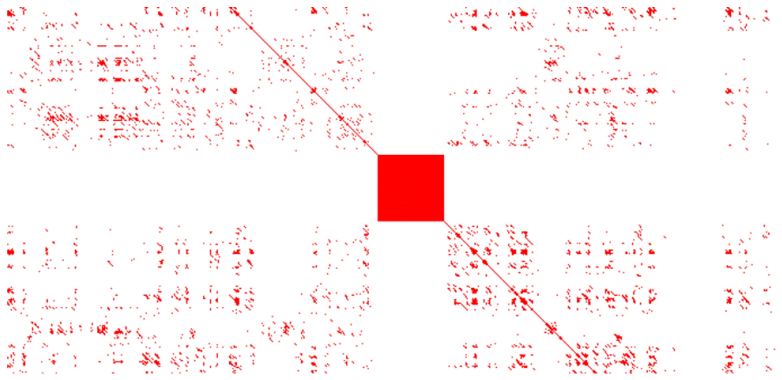


Figure 14.6. Dot matrix with tandem repeat.

The structural scheme of the method is as follows:

(1) Pre-processing character sequence. At this stage, the formation of the initial alphabet takes place: the removal of unnecessary characters and the recoding of the characters of the sequence.
(2) The transformation of the symbol sequence into a bundle of continuous characteristic functions based on the proved theorem.
(3) Conversion of characteristic functions into a spectral representation. Unlike the previous steps, this stage implies irreversible data compression.
(4) Spectral comparison of fragments of sequences.
(5) Display and analysis of the dot matrix in order to identify extended repeats and tandem repeats and study the relative position of repeats.
(6) Verification of repetitions by aligning with dynamic programming methods.

14.8 Recognition of Structural Motifs of α-α-Corner Type in Globular Proteins

One of the most frequently encountered structural motifs in homologous and nonhomologous proteins is α-α–corner [22] (Figure 14.7). This super-secondary structure is formed by two α-helices adjacent along a polypeptide chain, connected together by constrictions and

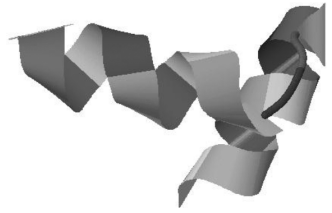


Figure 14.7. Tape model of α-α-corner with a short connection.

packed orthogonally. In proteins, α-α-corners are found in the form of a left super-helix. Their sequences have a specific arrangement in the chain of hydrophobic, hydrophilic, and glycine residues. In this chapter, we solve the problem of recognizing super-secondary structures in the PDB [6] database using a given template of the standard by the method of spectral-analytical comparison of the spatial structures of protein molecules [18, 50].

Within the framework of the task, it is necessary to form a sample of α-α-corners from the PDB database. The sample is a list of proteins with the coordinates of the atoms forming these structures.

The spatial structure of secondary and super-secondary structures in proteins is determined by the coordinates of the C_α atoms of the main chain of the protein globule. From this, it follows that when solving this problem, the coordinates of the atoms of the side chains of protein molecules can be neglected.

The spatial structure of the protein as a whole, as well as the spatial structure of the secondary and super-secondary structures, can be represented as a parametric curve equation in three-dimensional space:

$$\left\{ x(t) = \sum_{i=0}^{N} A_i \varphi_i(t), \quad y(t) = \sum_{i=0}^{N} B_i \varphi_i(t), \quad z(t) = \sum_{i=0}^{N} C_i \varphi_i(t), \right. \tag{14.14}$$

where $\{\varphi_i\}$ is a system of orthogonal polynomials and A_i, B_i, C_i are the coefficients of the expansion of functions in orthogonal bases. Curve (14.14) was obtained using the methods of analytical description of the main chain formed by the coordinates of C_α-atoms. Splines and orthogonal polynomials were used as such methods [16, 17]. The

Legendre and Chebyshev polynomials are most suitable for the functions under study. The decomposition coefficients, which are considered as the spectral characteristics of the signal, are calculated by the general formula

$$Z_i = \int_a^b f(t)\varphi_i(t)\rho(t)dt \tag{14.15}$$

with given weight $\rho(t)$.

Figure 14.7 shows the spatial structure of the classical α-α–corner, and Figure 14.8 its analytical description.

The transition from the parametric equation of a curve in three-dimensional space to a natural equation of a curve in three-dimensional space gives a parametrized description of the curvature and torsion depending on the natural parameter of the curve — the length of the arc (14.16):

$$\begin{cases} x = x(t) \\ y = y(t) \\ z = z(t) \end{cases} \Rightarrow \begin{cases} C(s) = \sqrt{\ddot{x}^2 + \ddot{y}^2 + \ddot{z}^2} \\ T(s) = \det \begin{vmatrix} \dot{x} & \dot{y} & \dot{z} \\ \ddot{x} & \ddot{y} & \ddot{z} \\ \dddot{x} & \dddot{y} & \dddot{z} \end{vmatrix} \cdot (\ddot{x}^2 + \ddot{y}^2 + \ddot{z}^2)^{-1}, \end{cases} \tag{14.16}$$

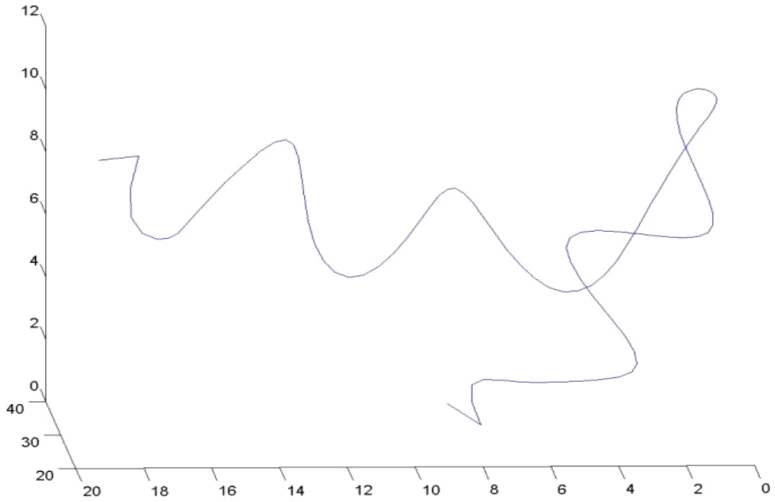


Figure 14.8. Analytical description of α-α-corner with a short connection by the Legendre polynomials.

where C is the curvature function, T is the torsion function, and s is the natural parameter of the curve is the arc length:

$$s(t) = \int_{t_1}^{t_2} \sqrt{\dot{x}^2 + \dot{y}^2 + \dot{z}^2}. \tag{14.17}$$

Such a description is invariant with respect to the choice of a Cartesian coordinate system. The functions of curvature and torsion are characteristic profiles of the spatial structure of the protein molecule. At the same time, regular segments, α-helix, are represented as regions of constant values of curvature and torsion. For example, Figure 14.9 shows the torsion profiles obtained from an analytical description of the structure of the α-α–corner presented in Figure 14.7.

For the given profiles, it is possible unambiguously to restore the spatial structure of the sample in Figure 14.7 with accuracy to the choice of the reference system of Cartesian coordinates. The transition from the natural equation of a curve in three-dimensional space to the parametric equation of a curve in three-dimensional space is carried out according to the Frenet formula [10].

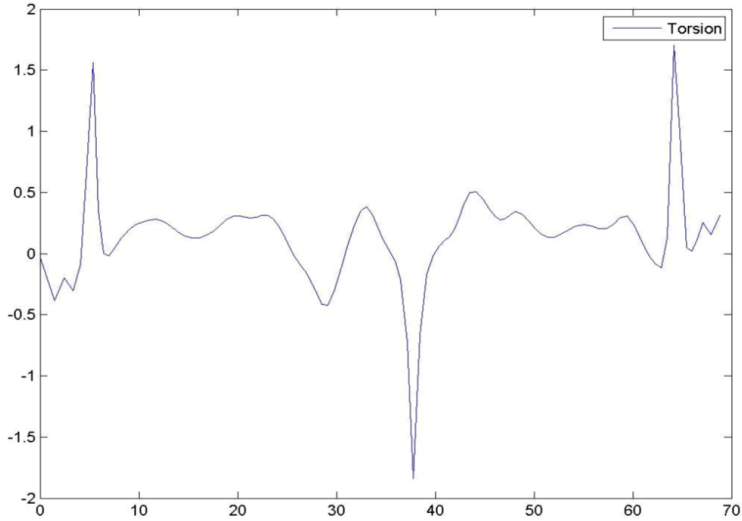


Figure 14.9. The torsion profile obtained from the analytical description of the α-α-corner presented in Figure 14.7.

At the first stage of training sample preparation, 110 α-α-corners were found in the PDB database corresponding to the specified standard 1D1L C_α: 15–37. For recognition, an approach was proposed based on the continual description of the protein molecule chain and the spectral method of repeating recognition.

The algorithm for searching α-α-corner in protein molecules is based on a generalized spectral-analytical method; it was developed and was previously successfully used to solve the problem of searching for repeats in genomic sequences [18, 50].

The approach described above is implemented in the form of the software-algorithmic complex "ProteinReviser". The search for α-α-corners was carried out according to the standard presented in Figure 14.7, in the PDB database. Using the developed method, 110 α-α-corners were found in the PDB database corresponding to a given standard 1D1L C_α: 15–37.

To solve the problem of recognition of structural motifs of proteins (by the example of α-α-corners), a combined approach is proposed for analyzing the spatial structure of proteins based on an analytical description of the main chain of the protein globule and spectral method of repeating recognition. In further studies by molecular dynamics was shown the stability of structural motifs, a comprehensive analysis of the geometric characteristics of these structures [57, 64, 65].

References

[1] Ahlberg, J.H., Nilson, E.N., and Walsh, J.L. (1967) *The Theory of Splines and Their Application* (Academic Press Inc., New York).

[2] Antoniou, A. (2006) Digital Signal Processing: Signals, Systems, and Filters (McGraw-Hill Professional, New-York), p. 965.

[3] Benson, G. (1999) Tandem repeats finder: A program to analyze DNA sequences. *Nucleic Acids Research*, 27, 573–578.

[4] Batenburg, K.J. and Sijbers, J. (2009) Adaptive thresholding of tomograms by projection distance minimization. *Pattern Recognition*, 42(10), 2297–2305.

[5] Belongie, S., Malik, J., and Puzicha, J. (2002) Shape matching and object recognition using shape contexts. *Transactions on Pattern Analysis and Machine Intelligence*, 24(4), 509–522.

[6] Berman, H.M., Westbrook, J., Feng, Z., Gilliland, G., Bhat, T.N., Weissig, H., Shindyalov, I.N., and Bourne, P.E. (2000) The protein data bank. *Nucleic Acids Research*, 28, 235–242.

[7] Bowmaker, J.K. (1998) Evolution of color vision in vertebrates. *Eye*, 12(3b), 541–547.

[8] Brazhnikov, E.V. and Efimov, A.V. (2001) Structure of α-α-hairpins with short connections in globular proteins. *Molecular Biology*, 35(1), 89–97.

[9] Britenkov, A.K. and Pankratov, A.N. (2004) Stable algorithms of adaptive approximation for acoustic signals description by orthogonal polynomials. *Physics of Wave Phenomena*, 12(3), 168–174.

[10] Bronshtein, I.N. and Semendyayev, K.A. (1986) *Handbook of Mathematics for Engineers and Students of Technical Colleges* (Science, Moscow) (in Russian).

[11] De Winter, J. and Wagemans, J. (2004) Contour-based object identification and segmentation: Stimuli, norms and data, and software tools. *Behavior Research Methods, Instruments, & Computers*, 36, 604–624.

[12] Dedus, F.F., Dedus, A.F., and Ustinin, M.N. (1992) A new data processing technology for pattern recognition and image analysis problems. *Pattern Recognition and Image Analysis*, 2, 195–207.

[13] Dedus, A.F., Dedus, F.F., Makhortykh, S.A., and Ustinin, M.N. (1993) Analytical description of multidimensional signals for solving problems of pattern recognition and image analysis. *Pattern Recognition and Image Analysis*, 3, 459–469.

[14] Dedus, A.F., Dedus, F.F., Makhortykh, S.A., and Ustinin, M.N. (1995) Generalized spectral-analytic method. Part I. Theoretical foundations. *Proceedings of SPIE*, 2363, 109–112.

[15] Dedus, A.F., Dedus, F.F., Makhortykh, S.A., and Ustinin, M.N. (1995) Generalized spectral-analytic method. Part II. Applications. *Proceedings of SPIE*, 2363, 109.

[16] Dedus, F.F., Makhortykh, S.A., Ustinin, M.N., and Dedus, A.F. (1999) *Generalized Spectral-Analytical Method for Processing Information Arrays* (Mashinostroenie, Moscow) (in Russian).

[17] Dedus, F.F., Kulikova, L.I., Pankratov, A.N., and Tetuev, R.K. (2004) *Classical Orthogonal Bases in the Problems of Analytical Description and Processing of Information Signals* (Department of Computational Mathematics and Cybernetics, Lomonosov Moscow State University, Moscow) (in Russian).

[18] Dedus, F.F., Kulikova, L.I., Makhortykh, S.A., Nazipova, N.N., Pankratov, A.N., and Tetuev, R.K. (2006) Analytical recognition methods for repeated structures in genomes. *Doklady Mathematics*, 74(3), 926–929.

[19] Dedus, F.F., Kulikova, L.I., Makhortykh, S.A., Nazipova, N.N., Pankratov, A.N., and Tetuev, R.K. (2007) Recognition of the structural–functional organization of genetic sequences. *Moscow University Computational Mathematics and Cybernetics*, 31(2), 49–53.

[20] Derguzov, A.V. and Makhortykh, S.A. (2006) Spectral analysis and data classification in magnetoencephalography. *Pattern Recognition and Image Analysis*, 16, 497–505.

[21] Du, L., Zhou, H., and Yan, H. (2007) OMWSA: Detection of DNA repeats using moving window spectral analysis. *Bioinformatics*, 23(5), 631–633.

[22] Efimov, A.V. (1984) A new super-secondary protein structure: The alpha alpha-angle. *Molecular Biology*, 18, 1524–1537 (in Russian).

[23] Efimov, A.V. (1993) Standard structures in proteins. *Progress in Biophysics and Molecular Biology*, 60, 201–239.

[24] Efimov, A.V. (2013) Super-secondary structures and modeling of protein folds. In Kister, A.E. (ed.) Methods in Molecular Biology, Vol. 932 (Humana Press, Clifton), pp. 177–189.

[25] Erdelyi, A. *et al.* (1953) *Higher Transcendental Functions*, Vol. 2, Chapter 10 (McGraw-Hill Book Co., New York).

[26] Florinsky, I.V. and Pankratov, A.N. (2015) Digital terrain modeling with orthogonal polynomials. *Machine Learning and Data Analysis*, 1(12), 1647–1659.

[27] Florinsky, I.V. and Pankratov, A.N. (2016) A universal spectral analytical method for digital terrain modeling. *International Journal of Geographical Information Science*, 30(12), 2506–2528.

[28] Frucci, M. and Sanniti di Baja, G. (2008) From segmentation to binarization of gray-level images. *Journal of Pattern Recognition Research*, 3(1), 1–13.

[29] Fukunaga, K. and Koontz, W.L.G. (1970) Application of the Karhunen-Loeve expansion to feature selection and ordering. *IEEE Transactions on Computers*, C-19(4), 311–318.

[30] Goncharov, V.L. (1954) *Theory of Interpolation and Approximation of Functions* (State Technical-Theoretical Publishing House, Moscow-Leningrag) (in Russian).

[31] Gordeev, A.B., Kargatov, A.M., and Efimov, A.V. (2010) PCBOST: Protein classification based on structural trees. *Biochemical and Biophysical Research Communications*, 397, 470–471.

[32] Katoh, K., Misawa, K., Kuma, K., and Miyata, T. (2002) MAFFT: A novel method for rapid multiple sequence alignment based on fast Fourier transform. *Nucleic Acids Research*, 30(14), 3059–3066.

[33] Kier, L., Seybold, P., and Cheng, C. (2005) *Modeling Chemical Systems Using Cellular Automata* (Springer, Netherlands).

[34] Kolpakov, R., Bana, G., and Kucherov, G. (2003) mreps: Efficient and flexible detection of tandem repeats in DNA. *Nucleic Acid Research*, 31, 3672–3678.

[35] Kovalevsky, V. (2004) Algorithms in digital geometry based on cellular topology. *Proceedings of IWCIA: Combinatorial Image Analysis*, pp. 366–393.

[36] Krumsiek, J., Arnold, R., and Rattei, T. (2007) Gepard: A rapid and sensitive tool for creating dotplots on genome scale. *Bioinformatics*, 23(8), 1026–1028.

[37] Landau, G.M., Schmidt, J.P., and Sokol, D. (2001) An algorithm for approximate tandem repeats. *Journal of Computational Biology*, 8, 1–18.

[38] Loginov, V.M., Tsepkov, G.V., and Chinaev, P.I. (1976) *Economical Coding* (Tekhnika, Kiev) (in Russian).

[39] Makhortykh, S.A. and Semechkin, R.A. (2008) Spectral methods of magnetic encephalogram analysis, dynamics of inhomogeneous systems. *Proceedings of the Institute for System Analysis of the Russian Academy of Sciences*, 33, 236–250.

[40] Makhortykh, S.A. and Semechkin, R.A. (2009) Brain magnetic field study for Parkinson decease. *Bulletin of Experimental Biology and Medicine*, 147, 351.

[41] Makhortykh, S.A. and Semechkin, R.A. (2009) Localization and brain biomagnetic activity study for auditory stimulation. *Bulletin of Experimental Biology and Medicine*, 147, 477.

[42] Makhortykh, S.A. (2016) Generalized spectral-analytical method for biomedical data processing. *Mathematica Montisnigri*, 36, 104–113.

[43] Mingqiang, Y., Kpalma, K., and Ronsin, J. (2008) A survey of shape feature extraction techniques. *Pattern Recognition by Peng-Yeng Yin* (In-Tech).

[44] Mobahi, H., Rao, S.R., Yang, A.Y., *et al.* (2011) Segmentation of natural images by texture and boundary compression. *International Journal of Computer Vision*, 95(1), 86–98.

[45] Nikiforov, A.F. and Uvarov, V.B. (1988) *Special Functions of Mathematical Physics* (Birkhauser, Basel).

[46] Nikiforov A.F., Uvarov V.B., and Suslov S.K. (1991) *Classical Orthogonal Polynomials of a Discrete Variable.* Springer Series in Computational Physics (Springer, Berlin).

[47] Sarvas, J. (1987) Basic mathematical and electromagnetic concepts of the biomagnetic inverse problem. *Physics in Medicine and Biology*, 32, 11–22.

[48] Ogurtsov, A.Y., Roytberg, M.A., Shabalina, S.A., and Kondrashov, A.S (2002) OWEN: Aligning long collinear regions of genomes. *Bioinformatics*, 18, 1703–1704.

[49] Pankratov A.N. (2004) On the implementation of algebraic operations on orthogonal function series. *Computational Mathematics and Mathematical Physics*, 44(12), 2017–2023.

[50] Pankratov, A.N., Gorchakov, M.A., Dedus, F.F., Dolotova, N.S., Kulikova, L.I., Makhortykh, S.A., Nazipova, N.N., Novikova, D.A., Olshevets, M.M., Pyatkov, M.I., Rudnev, V.R., Tetuev, R.K., and Filippov, V.V. (2009) Spectral analysis for identification and visualization of repeats in genetic sequences. *Pattern Recognition and Image Analysis*, 19(4), 687–692.

[51] Pankratov, A.N., Pyatkov, M.I., Tetuev, R.K., Nazipova, N.N., and Dedus, F.F. (2012) Search for extended repeats in genomes based on the spectral-analytical method. *Mathematical Biology & Bioinformatics*, 7(2), 476–492.

[52] Pankratov, A.N., Tetuev, R.K., Pyatkov, M.I., Toigildin, V.P., and Popova, N.N. (2015) Spectral analytical method of recognition of inexact repeats in character sequences. *Proceedings of the Institute for System Programming*, 27(6), 335–344.

[53] Pankratov, A.N. and Pankratova, N.M. (2022) Spectral method for detecting inexact repeats in character sequences. *Pattern Recognition and Image Analysis*, 32(3), 622–625.

[54] Pham, M.-T., Gao, Y., Hoang V., and Cham, T. (2010) Fast polygonal integration and its application in extending Haar-like features to improve object detection. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, CVPR*, pp. 942–949.

[55] Pyatkov, M.I., Filippov, V.V., and Pankratov, A.N. (2012) Consensus of repeated region of rabbit chromosome 17 containing over 15 huge approximate tandem repeats. *Repbase Reports*, 12(3), 256–256.

[56] Pyatkov, M.I. and Pankratov, A.N. (2014) SBARS: Fast creation of dotplots for DNA sequences on different scales using GA-, GC-content. *Bioinformatics*, 30(12), 1765–1766.

[57] Rudnev, V.R., Pankratov, A.N., Kulikova, L.I., Dedus, F.F., Tikhonov, D.A., and Efimov, A.V. (2013) Recognition and Stability analysis of structural motifs of α-α-corner type in globular proteins. *Mathematical Biology & Bioinformatics*, 8(2), 398–406.

[58] Sharma, D., Issac, B., Raghava, G.P.S., and Ramaswamy, R. (2004) Spectral repeat finder (SRF): Identification of repetitive sequences using Fourier transformation. *Bioinformatics*, 20(9), 1405–1412.

[59] Tetuev, R.K. (2007) Algebra of Spectral Transformations. Doctoral dissertation, Dorodnicyn Computing Centre of the Russian Academy of Sciences, Moscow, Russia.

[60] Tetuev, R.K., Nazipova, N.N., Pankratov, A.N., and Dedus, F.F. (2010) Search for megasatellite tandem repeats in eukaryotic genomes

by estimation of GC-content curve oscillations. *Mathematical Biology & Bioinformatics*, 5(1), 30–42.

[61] Tetuev, R.K. and Nazipova, N.N. (2010) Consensus of repeated region of mouse chromosome 6 containing 60 tandem copies of a complex pattern. *Repbase Reports*, 10(5), 776.

[62] Tetuev, R.K., Dedus, F.F., and Nazipova, N.N. (2010) Consensus of repeated region of rat chromosome 4 similar to mouse chromosome 6 repeated region, enclosed in the intergenic region between genes Hrh1 and Atg7. *Repbase Reports*, 10(8), 1185.

[63] Tetuev, R.K., Pyatkov, M.I., and Pankratov, A.N. (2017) Parallel algorithm for global alignment of long aminoacid and nucleotide sequences. *Mathematical Biology & Bioinformatics*, 12(1), 137–150.

[64] Tikhonov, D.A., Kulikova, L.I., and Efimov, A.V. (2016) Statistical analysis of the internal distances of helical pairs in protein molecules. *Mathematical Biology & Bioinformatics*, 11(2), 170–190.

[65] Tikhonov, D.A., Kulikova, L.I., and Efimov, A.V. (2018) Analysis of torsion angles between helical axes in pairs of helices in protein molecules. *Mathematical Biology & Bioinformatics*, 13(S), t17–t28.

[66] Ustinin, M.N., Makhortykh, S.A., *et al.* (2002) Problems of magnetoencephalography data analysis. In Lakhno, V.D. and Ustinin, M.N. (eds.) *Computers and Supercomputers in Biology* (Institute of Computer Science, Moscow), pp. 327–349 (in Russian).

[67] Vilenkin, N.J. (1968) Special Functions and Theory of Group Representations (AMS, New York).

[68] Watanabe, S. (1965) Karhunen-Loeve expansion and factor analysis. *Transactions of 4th Prague Conference on Information Theory.*

[69] Žalik, B., Mongus D., Liu Y.-K., and Lukač, N. (2016) Unsigned Manhattan chain code. *Journal of Visual Communication and Image Representation*, 38, 186–194.

© 2025 World Scientific Publishing Company
https://doi.org/10.1142/9789811267215_00015

Chapter 15

Recognition of Biomedical Signals for Medical Diagnostics

Anatoly P. Nemirko* **and Ludmila A. Manilo†**

Biotechnical Systems Department at Saint-Petersburg Electrotechnical University "LETI", St. Petersburg, the Russian Federation

**apn-bs@yandex.ru*

†*lmanilo@yandex.ru*

In this chapter, a number of methods for biomedical signal recognition in specific tasks of medical diagnostics are considered. All of them are united by the need to search non-standard approaches for the solution of practical tasks. It is connected with small distinctions of classes in initial multidimensional space, and with specific properties of the recognizable signals which are difficult for revealing by traditional temporary and frequency methods.

The method of recognition of signals with the expressed irregular component by Shannon conditional entropy parameters is considered. A number of parameters of conditional entropy are investigated on model signals. This allowed choosing an assessment of signal irregularity degree and successfully applying it for heart rhythm abnormalities (atrial fibrillation) detection on the sequence of intervals between heart beats. Also interesting task in the field of medical diagnostics devoted to the recognition of signals on the intensity of their nonlinear properties is considered. We used the approach based on the use of approximated entropy which is closely connected with Kolmogorov's entropy (K-entropy). The algorithm for the calculation of this characteristic and its main properties and parameters is considered in detail. They can be used for the dynamic irregularities detection connected with nonlinear properties of signals. Examples of the detection of cardiac fibrillation problems

using the parameters of heart rhythm nonlinearity and also an anesthesia depth assessment according to the electroencephalogram are given.

The method of electrocardiogram signal recognition according to its spectral description on the base of the multiple discriminant analysis is considered in detail. It is shown that the use of Fischer's criterion with the introduction of weight functions, allows increasing the accuracy of signal classification. Examples of recognition of dangerous arrhythmias, including detection of ventricular fibrillation of the heart at the early stages of its emergence are given. The technique of dangerous arrhythmias recognition according to the electrocardiogram description in the frequency domain is offered. Ways of the spectral description forming and methods of linear decision functions creation by Fischer's criterion are considered. The experiments carried out showed high efficiency of the offered algorithms.

Keywords: recognition of biomedical signals, Shannon conditional entropy, Kolmogorov entropy, approximate entropy, chaotic properties, nonlinear components, spectral analysis, signal classification, linear discriminant analysis, Fisher's criterion, multiclass problem, medical diagnostic, cardiac monitoring systems, atrial fibrillation recognition, estimates of the depth of anesthesia by the electroencephalogram.

15.1 Introduction

Today, computers have become an integral component of systems for analyzing biomedical signals. They perform various tasks from data collection and preliminary processing to the detection and interpretation of diagnostic features. Analysis of signals is not always a simple task for a physician or life-sciences specialist. Clinically relevant information in the signal is often masked by noise and interference. In addition, the signal parameters may not be directly perceived by the visual or auditory systems of a human observer. The form of surface electroencephalographic (EEG) or electromyographic (EMG) signal is too complex for visual analysis. Some tasks that require continuous attention, such as the routine ECG monitoring of patients with severe cardiological conditions or the EEG monitoring of anesthesia depth during surgical operations, require the long-term work of highly qualified personnel. These factors determine the need for developing methods for objective signal analysis using processing algorithms implemented on computer systems.

The development of algorithms for the analysis of biomedical signals is a challenge because of the variability and diversity of features of biomedical signals, where these problems are manifested much more often than in physical systems. However, the analysis of biomedical signals has currently reached the level of practical application of signal processing and pattern recognition methods for effective noninvasive diagnostics, regular condition monitoring of seriously ill patients, and rehabilitation and sensory support for the disabled.

In this chapter, a number of methods for detecting biomedical signals for specific problems that are relevant in the field of medical diagnostics are considered. All of them are united by the necessity of searching for nonstandard approaches for solving practical problems.

Section 15.2 deals with the method for detecting signals with a pronounced irregular component by the Shannon conditional entropy parameters. A number of conditional entropy parameters are studied on model signals. As a result, the signal irregularity degree is estimated and successfully applied to detecting atrial fibrillation areas in the analysis of rhythmograms.

Section 15.3 is devoted to the recognition of signals by the degree of manifestation of their nonlinear components. In this case, an approach based on the use of approximate entropy closely related to the Kolmogorov entropy (K-entropy) is used. An algorithm for calculating this characteristic is considered in detail. Its parameters can be used to detect dynamic irregularities associated with nonlinear signal properties. Examples of solving the problems of detecting atrial fibrillation by the rhythmogram nonlinearity parameters and the estimates of the depth of anesthesia by the electroencephalogram are given.

Section 15.4 proposes a method for detecting dangerous arrhythmias by describing the ECG in the frequency domain. Methods for generating spectral descriptions and methods for constructing linear decision functions by Fisher's criterion are discussed. The conducted experiments show high effectiveness of the proposed algorithms.

In Section 15.5, the method for recognizing different forms of an electrocardiosignal by spectral description using multiple discriminant analysis is considered in detail. It is shown that the use of Fisher's criterion with the introduction of weight functions makes it possible to improve the signal classification accuracy. Examples of

solving the problems of recognition of dangerous arrhythmias including the detection of ventricular fibrillation of the heart at its early stages are given.

Many of the examined algorithms were tested on real signals and used in developing practical systems for health care.

15.2 Analysis of Complex Biosignals using Conditional Entropy Estimates

A number of problems in the analysis of biomedical signals involve detecting fragments of finite length by the degree of manifestation of deterministic and random components for detecting irregular discrete sequences. It can be an analysis of heart rhythm variability in different states of a living organism, recognition of atrial fibrillation attacks, etc. One of the methods for solving them is the approach based on Shannon's information theory [45].

The regularity of the time series converted by quantization and character encoding into a sequence of states is understood as the statistical stability of the transition structure observed over time. Conditional entropy can be used as an indicator of the emergence of new sequences that are inconsistent with the preceding ones [10, 11, 42]. An analysis of the change in this index over time, i.e. with an increase in the length of the observed sequences, can serve as an effective estimate of the time series irregularity.

15.2.1 *Calculation and selection of conditional entropy parameters*

First, two parameters are given: L is the length of the sample sequence and ξ is the number of interval estimates of discrete samples of the signal $x(i), i = 1, 2, \ldots, N$ in the range $(x_{\min}, x_{\max})$. The procedure for finding the conditional entropy index involves the following sequence of steps:

(1) Generation of $(L-1)$-dimensional sequences for a given sample:

$$X_{L-1}(i) = x(i), x(i+1), \ldots x(i+L-2),$$
$$i = 1, 2, \ldots, (N-L+2).$$

(2) Their representation in the form of strings of symbols: $Z_{L-1}(i), i = 1, 2, \ldots, (N - L + 2)$ obtained by level quantization and coding of the resulting values with symbols from an arbitrary alphabet $A = \{a_p\}, p = 1, \ldots, \xi$ (usually $\xi = 6{-}8$).

(3) Calculation of the value of unconditional entropy:

$$E(L-1) = -\sum_{L-1} p_{L-1} \log p_{L-1}, \tag{15.1}$$

where p_{L-1} is the probability of the occurrence of a specific string of symbols with length $L-1$ (in practical calculations, the probability estimate is the frequency at which the string of symbols ends up in the corresponding cell of the $(L-1)$-dimensional space).

Here, the summation is over all observed types of strings of symbols with lengths of $L-1$. Each string type is distinguished by a nonrepeating sequence of events (symbols).

(4) Transition to the entropy estimate for L-point sequences. Repetition of Steps 1–3 with the calculation of $E(L) = -\sum_L p_L \log p_L$ for the whole set of states.

(5) Calculation of the value of the conditional entropy $E(L|L-1)$ in the form of an increment obtained by the unconditional entropy $E(L)$ upon passing from the sequence of events with length $L-1$ to the length L:

$$E(L|L-1) = E(L) - E(L-1) = -\sum_{L} p_L \log p_L + \sum_{L-1} p_{L-1} \log p_{L-1}. \tag{15.2}$$

For long samples ($N \to \infty$), the value of $E(L|L-1)$ tends to zero as L increases for regular processes and to a constant positive value for processes with purely random properties. However, the replacement of the probabilities in (15.1) and (15.2) by the values of the corresponding frequencies leads to a significant shift in the estimate of $E(L|L-1)$ in the analysis of a finite sample. This is due to the fact that as L increases, the number of strings that were encountered only once increases, which gives zero excess entropy. Hence, false regularity can be detected even for a purely random process, because the conditional entropy will tend to zero. In fact, such single strings

are "unknown" in the sense of the predictability of behavior dynamics in subsequent steps $(L, L+1, L+2, \ldots)$. The following types of correction can be used to exclude the false regularity introduced into the estimate of conditional entropy by single strings:

$$\tilde{E}_1(L|L-1) = E(L|L-1) \cdot (1 + N^{(1)}_{L-1}/(N_L - N^{(1)}_{L-1})), \quad (15.3)$$

$$\tilde{E}_2(L|L-1) = E(L|L-1) + E(1) \cdot N^{(1)}_{L-1}/N_L, \quad (15.4)$$

where N_L is the number of analyzed strings of symbols with length $L, N^{(1)}_{L-1}$ is the number of single strings with length $(L-1)$, and $E(1) = E(L), L = 1$ is the entropy of single events.

Here, the corrected estimates of the conditional entropy $\tilde{E}_1(L|L-1)$ and $\tilde{E}_2(L|L-1)$ differ from the conditional entropy $E(L|L-1)$ by some increments that are given to single strings. In Eq.(15.3), the entropy increment for $N^{(1)}_{L-1}$ single strings is given as the mean increment of the conditional entropy $E(L|L-1)$ estimated by $(N_L - N^{(1)}_{L-1})$, which is the number of sequences of events. In formula (15.4), such events are assigned weights $E(1)$ corresponding to the entropy of single events. Due to this, "unknown" strings are considered as random and not regular sequences. It makes it possible to avoid the false representation of unconfirmed regular changes in the examined sequences. The introduction of the weights given by $E(1)$ was justified in [42], but they were given to all single events found at the Lth step, which led to entropy displacement. In [10, 11], only the strings that are represented by only one object in the cells of the $(L-1)$-dimensional state space were assigned to the category of completely random events.

Figure 15.1 shows the dependences of the original $E(L|L-1)$ and corrected $\tilde{E}_1(L|L-1)$ and $\tilde{E}_2(L|L-1)$ estimates of the conditional entropy on the length of the strings L for three types of model signals: (a) a harmonic signal, (b) noise, and (c) an additive mixture of signal and noise. The comparative analysis of the initial and corrected curves indicates that for the harmonic signal (Figure 15.1(a)), their changes almost coincide, and if there is noise (Figures 15.1(b) and 15.1(c)), they differ significantly. These differences are associated with the conditional entropy correction only in the number of single strings, which is $N^{(1)}_{L-1}$. At the same time, the estimate $\tilde{E}_1(L|L-1)$ slightly reduces the change rate of conditional entropy,

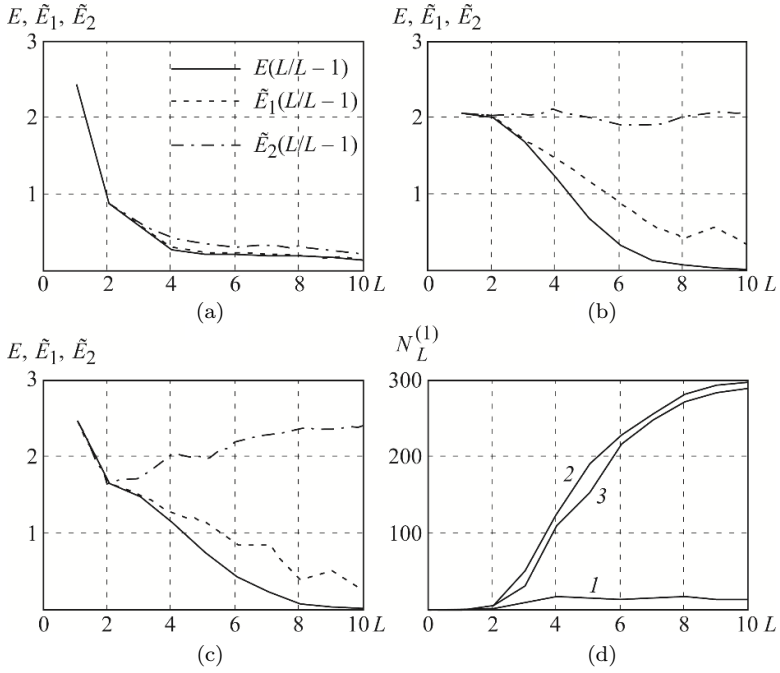


Figure 15.1. Dependences of conditional entropy estimates $E(L|L-1)$, $\tilde{E}_1(L|L-1), \tilde{E}_2(L|L-1)$ for different types of model signals.

and the estimate $\tilde{E}_2(L|L-1)$ approaches the constant level given by $E(1)$, which can be interpreted as an artificial noise contamination of the analyzed signal. Figure 15.1(d) shows the dependence of $N_L^{(1)}$ on the length L for a harmonic signal (curve 1), noise (2), and a mixture of harmonic signal and noise (3). It can be seen from the figure that for a sample of bounded length N, the number of "unknown" strings increases rapidly with increasing L if there is a random component in the signal, which requires correction of the function $E(L|L-1)$.

The following signal recognition parameters were selected based on the analysis of the obtained dependencies. First, these can be point estimates of the corrected curve $\tilde{E}_1(L|L-1)$ for values of $L = 2, 3, 4$, i.e. in the region where the contribution of single strings is negligible. It makes it possible to use them as discriminant features in solving signal classification problems. Second, the lower boundary of the curve $\tilde{E}_2(L|L-1$ [42] can be used as a process irregularity measure, which assumes a small value for regular processes and a large value in

the case of pronounced random changes. The relative minimum ME given in the form M$E = E(1) - \min_{L=2,\ldots,10}\{\tilde{E}_2(L|L-1)\}$ was calculated for this purpose in [10,11]. The higher the process regularity, the greater the index value ME.

Experiments on real data made it possible to establish that the function $\tilde{E}_1(L|L-1)$ and the index ME can be used to estimate the degree of regularity of changes observed in time series of finite length and can also be used in the problem of recognition of rhythm disturbances with a complex dynamics of the distribution of cardio intervals.

15.2.2 *Detection of atrial fibrillation fragments*

The estimate ME was used in the problem of atrial fibrillation recognition by the sequence of cardiocycles with length $N = 300$ [11,12]. More than 150 ECG records were processed during the research. Their rhythmograms were constructed, and the relative minimum of the $\tilde{E}_2(L|L-1)$ function was calculated. Analysis of the experimental results showed that the parameter ME has stable interval estimates for a significance level of $\alpha = 0.05$ for different types of considered rhythmograms, and the value of this parameter for atrial fibrillation is much smaller. Thus, the value of ME is (0.85 ± 0.09) for a normal rhythm with a wave rhythmogram structure. In the case of frequent premature ventricular contractions, which are characterized by short-term regular changes in the sequence of RR intervals, they are somewhat smaller than (0.74 ± 0.10). The value of ME is much smaller than (0.17 ± 0.11) only in the case of atrial fibrillation, which is characterized by an irregular distribution of cardiac cycle lengths.

The result of the dynamic analysis of the parameter ME for a complex sequence of RR intervals containing a short-term atrial fibrillation attack is given as an example in Figure 15.2. If the current processing window with length $N = 250$ covers the data area that contains chaotic changes in the rhythmogram, the value of ME sharply decreases, which indicates the arrhythmia attack.

It was established in the course of numerous models and real experiments that the index ME can be effectively used for the analysis of irregular sequences while being the integral parameter of the corrected conditional entropy. In particular, it can be used for the

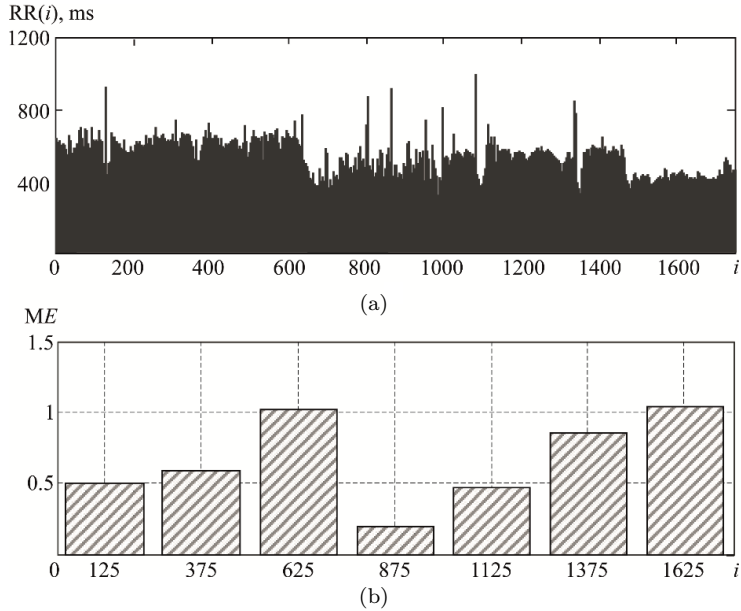


Figure 15.2. Analysis of the sequence of RR-intervals containing a segment of atrial fibrillation: (a) rhythmogram with severe atrial fibrillation (data area from 700 to 950); (b) the parameter calculation result ME.

detection of fragments of atrial fibrillation during the automatic monitoring of the patient's ECG.

15.3 Recognition of Biosignals with Nonlinear Properties by Approximate Entropy Parameters

Complex biomedical signals (EEG, ECG, heart rhythm, etc.) under certain conditions can be considered as the processes generated by the dynamic chaos mode. This representation makes it possible to diagnose using criteria based on the analysis of characteristics used in nonlinear dynamics. For example, in [1, 41], it was proposed to use *approximate entropy* (ApEn) which quantitatively determines the degree of signal complexity.

Approximate entropy is closely related to Kolmogorov entropy (K-entropy) [35, 44], the most important characteristic of chaotic motion in a phase space of arbitrary dimension. K-entropy is defined as the average rate of information loss about the state of the dynamic

system over time. It is zero for regular motion, infinite for random systems, and positive and constant for systems with deterministic chaos. By analyzing some signal that seems random, it is possible to distinguish the irregularity associated with the motion dynamics on the attractor from the white noise impact on the system [1].

In practice, the value of K-entropy is difficult to compute because one implementation of a complex process is analyzed. Although an attractor characterizing the properties of the system is embedded in the d-dimensional phase space. However, according to Takens' theorem [44], some properties of the attractor can be reconstructed by one process component. In this case, it is possible to find the lower bound of the K-entropy

$$K_2 = \lim_{r\to 0}\lim_{m\to\infty} \ln \frac{C_m(r)}{C_{m+1}(r)}, \tag{15.5}$$

where $C_m(r)$ is the generalized correlation integral, m is the length of a sequence of points, and r is the threshold distance.

The correlation integral is calculated as follows:

$$C_m(r) = \lim_{N\to\infty} \frac{1}{N^2} \sum_{i}^{N} \sum_{j}^{N} \Theta(r - |X(i) - X(j)|); i \neq j, \tag{15.6}$$

where $X(i) = (x(i), x(i+1), \ldots, x(i+m-1))$ is a sequence consisting of m signal samples, $\Theta(\cdot)$ is the Heaviside function ($\Theta(s) = 1$, if $s \geq 0$, and $\Theta(s) = 0$, if $s < 0$), and N is the data sample size.

If the index is $K_2 > 0$, it denotes a sufficient condition for chaos. However, the requirement $m \to \infty, r \to 0, N \to \infty$ in (15.5) and (15.6) makes it impossible to apply the K_2 criterion in practical problems of signal analysis on finite samples. Thus, it is necessary to find an approximate estimate of the Kolmogorov entropy in the form of ApEn.

15.3.1 *Algorithm for calculating the ApEn function*

Let there be a sample of the source data $x(1), x(2), \ldots, x(N)$, where N is the sample length. First, let us set the values of two parameters: m is the length of the analyzed strings and r is the threshold specifying the sizes of the cells of the phase space:

(A) Let us generate the sequences $X(1), \ldots, X(N-m+1)$ defined by the following expression:

$$X(i) = (x(i), x(i+1), \ldots, x(i+m-1)),\ i = 1, \ldots, (N-m+1).$$

(B) Determine the distances between $X(i)$ and $X(j)$ as follows:

$$d(X(i), X(j)) = \max_{k=0,\ldots,(m-1)} (|x(i+k) - x(j+k)|).$$

(C) Calculate

$$C_r^m(i) = N^m(i)/(N-m+1),$$

where $N^m(i)$ is the number of values of $d(X(i), X(j))$ that satisfy the following condition:

$$d(X(i), X(j)) \le r; \quad j = 1, \ldots, (N-m+1).$$

(D) Take the natural logarithm from each $C_r^m(i)$ and average its value over i:

$$\theta^m(r) = \frac{1}{N-m+1} \sum_{i=1}^{N-m+1} \ln C_r^m(i).$$

(E) Increase the value of m to m+1. Repeat steps 1–4 and find the values of $C_r^{m+1}(i)$, $\theta^{m+1}(r)$.

(F) Find the K-entropy estimate

$$\mathrm{ApEn}(m, r) = \lim_{N \to \infty} (\theta^m(r) - \theta^{m+1}(r)).$$

A high-speed algorithm for computing this ApEn estimate was proposed in [1].

The value of the ApEn depends on the parameters m and r. It is assumed [41] to set the values of $m = 2$ and r to $(0.1, \ldots, 0.25)\sigma_x$, where σ_x is the standard deviation of the original data sample. However, a more complete idea about the properties of the investigated signal can be obtained by analyzing the change in the entropy index over time, i.e. under successive increases in the length of the strings m. All subsequent calculations of the ApEn were carried out for $N = 300, m = 1, \ldots, 6$, and $r = 0.15\ \sigma_x$.

The correction estimate ApEn(m) can be used to exclude a false signal regularity estimate associated with the finite sample length N [12, 20, 29]:

$$\mathrm{ApEn}_{\mathrm{cor}}(m) = \mathrm{ApEn}(m) + \mathrm{ApEn}(0) \cdot N_m^{(1)}/N_{m+1}, \qquad (15.7)$$

where N_{m+1} is the number of the analyzed strings with length $(m+1)$, $N_m^{(1)}$ is the number of once encountered strings with length m, and ApEn(0) is the absolute entropy value.

In this case, the lower bound of the K-entropy can be approximated by the lower bound $\mathrm{ApEn}_{\mathrm{cor}}(m)$. Note that correction type (15.7) is similar to correction estimate (15.4) used in the Shannon conditional entropy analysis.

The main difference between the ApEn and the conditional entropy $E(L|L-1)$ (15.2) is the method for generating elementary regions of the phase space. Hypercubes are generated by a rigid grid given by the number of quantization levels ξ within the values of $(x_{\min}, x_{\max})$ in the calculation of conditional entropy. The mechanism of adaptive generation of elementary regions by constructing hyperspheres with a radius r with the center specified by each point in which the object of the phase space is located is active in calculating ApEn(m). A schematic depiction of the mechanisms of generation of elementary cells for a 2D space is shown in Figure 15.3.

15.3.2 *Analysis of properties and selection of parameters of ApEn*

Model experiments aimed at studying the properties of approximate entropy were conducted in [1, 20, 29]. The following characteristics

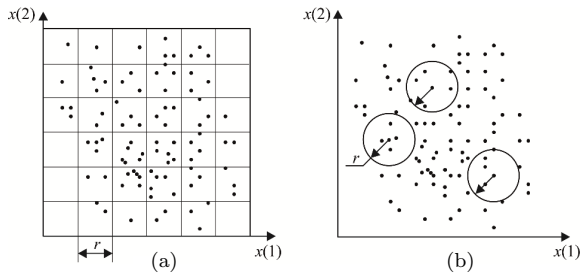


Figure 15.3. Illustration of the mechanism for the generation of phase space cells in the calculation of (a) conditional entropy; (b) ApEn.

can be distinguished from the main properties:

1. Approximate entropy reflects the signal complexity degree, i.e. the higher its regularity, the less the value of ApEn(m).
2. Analysis of ApEn(m) makes it possible to obtain a reliable estimate of the degree of chaos of the signal by relatively short signal strips.
3. Estimates ApEn(m) can be used to detect dynamic irregularities associated with both nonlinear and stochastic signal properties.
4. Estimates ApEn(m) are resistant to short-term noise.
5. Parameter r used in the calculation of ApEn(m) is also a filter parameter that suppresses noise.
6. Approximate entropy is insensitive to the change in the frequency of the harmonic signal.

Experiments showed that it is possible to detect signals in which their chaotic properties are of a different nature.

Figure 15.4 shows the result of analyzing the logistic mapping $X(i+1) = a \cdot X(i) \cdot (1 - X(i))$, where a is an external parameter, as an example. This nonlinear difference equation for $a = 3.5$ is periodic (Figure 15.4(a)), and for $a = 3.8$ it is chaotic (Figure 15.4(c)).

There are differences in the obtained characteristics in the graphs, namely: for $a = 3.5$, ApEn(m) rapidly decreases to zero with increasing m (Figure 15.4(b)), which is typical for periodic signals, and the correction does not change the estimate $\mathrm{ApEn}_{\mathrm{cor}}(m)$ (the graphs of the dependences coincide).

At the same time, the dependence ApEn(m) tends to a certain constant level with increasing m in the transition to the deterministic chaos mode ($a = 3.8$) (Figure 15.4(d)). In this case, at step $m = 5$, a correction is introduced.

The shape of the curves ApEn(m) and $\mathrm{ApEn}_{\mathrm{cor}}(m)$ changes abruptly in the analysis of a stochastic process. Figure 15.5 shows the result of the analysis of the second-order autoregression:

$$X(i) - \mu = a_1 \cdot (X(i-1) - \mu) + a_2 \cdot (X(i-2) - \mu) + Z(i),$$

where μ is the mathematical expectation of a random variable $Z(i), Z(i), i = 1, 2, \ldots$ is the white noise with normal distribution law, $a_1 = 1.0$, and $a_2 = -0.5$.

As can be seen from Figure 15.5(b), there is a pronounced minimum $\mathrm{ApEn}_{\mathrm{cor}}(m)$ in the graphs. The values of ApEn(m) tend to

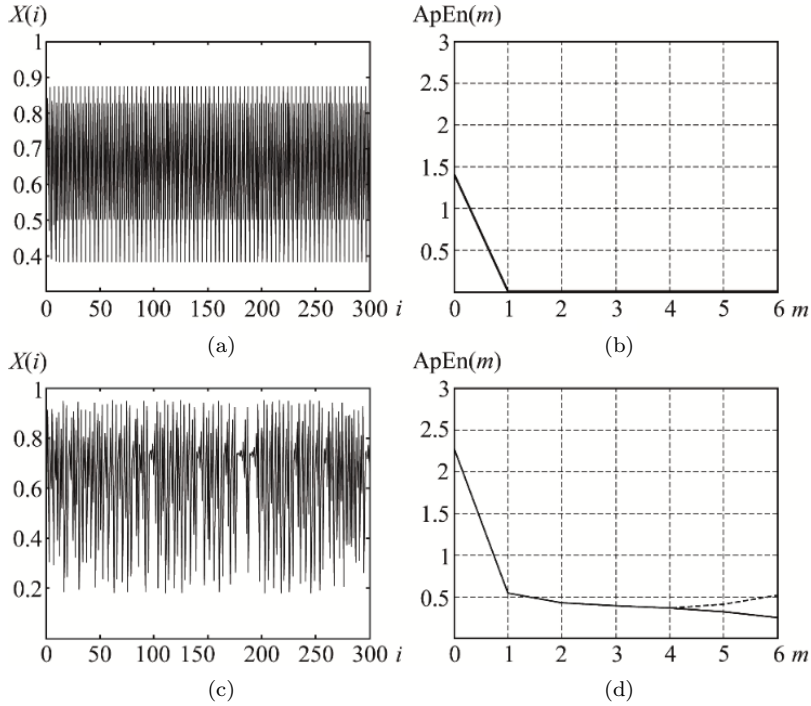


Figure 15.4. Logistic mapping for a = 3.5 and a = 3.8: (a, c) the original signal $X(i)$ and (b, d) dependences of ApEn(m) (solid line) and ApEn$_{\rm cor}(m)$ (dashed line) on m.

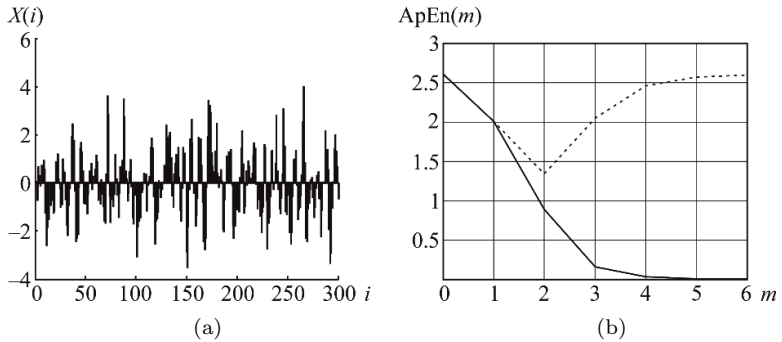


Figure 15.5. Second-order autoregressive process: (a) the original signal $X(i)$; (b) dependences of ApEn(m) (solid line) and ApEn$_{\rm cor}(m)$ (dashed line) on m.

zero, while the function $\mathrm{ApEn}_{\mathrm{cor}}(m)$ tends to the value of the absolute entropy ApEn(0). The nature of the dependences is similar to the laws found in the case of white noise.

The following parameters can be used to estimate the regularity degree of the changes observed in a discrete sequence of samples:

- the values of $\mathrm{ApEn}(m)$ and $\mathrm{ApEn}_{\mathrm{cor}}(m)$ for $m = 1, 2, 3$, where the contribution of single strings is negligible,
- relative minimum $\mathrm{M}E_{\mathrm{K}} = \mathrm{ApEn}(0) - \min\limits_{m=1\ldots6}\{\mathrm{ApEn}_{\mathrm{cor}}(m)\}$ which approximates the lower bound of the K-entropy.

15.3.3 *Atrial fibrillation recognition*

These parameters were used to recognize atrial fibrillation by the sequence of cardiointervals [12, 30]. The analysis of the correlation dimension of rhythmograms in the case of atrial fibrillation showed [31] that the heart rhythm is chaotic, and, therefore, methods of nonlinear dynamics can be used in this problem. The studies were performed on the rhythmograms of real electrocardiogram records. All the realizations were previously verified and divided into three classes: normal rhythm (NR) including pronounced respiratory arrhythmia, atrial fibrillation (AF), and frequent premature heartbeats (FPHB). The choice of these classes is associated with the possible erroneous recognition of AF strips for the alternative NR and FPHB groups which have a significant scatter in the rhythmogram readings. Then, the most characteristic realizations for this class (by 50 implementations with length 300 samples) were selected in each of the three groups, and training samples were generated. For these fragments, the values $\mathrm{ApEn}(m)$ and $\mathrm{ApEn}_{\mathrm{cor}}(m)$ were calculated for $m = 1, \ldots, 6$, and also $\mathrm{M}E_K$.

Figure 15.6 shows examples of rhythmograms and the obtained dependences of $\mathrm{ApEn}(m)$ on the fragment lengths m for the classes AF (a, b), NR (c, d), and FPHB (e, f) for $r = r_0 \cdot \sigma_x, r_0 = 0.15$. As can be seen from the figure, the entropy dependencies have different forms for different classes of rhythmograms. In the case of AF, which is characterized by a chaotic rhythmogram, there is a decrease and then a rapid return of the corrected entropy to the original level ApEn(0). As a consequence of this change in $\mathrm{ApEn}_{\mathrm{cor}}(m)$, the parameter $\mathrm{M}E_{\mathrm{K}}$ for AF takes the smallest value.

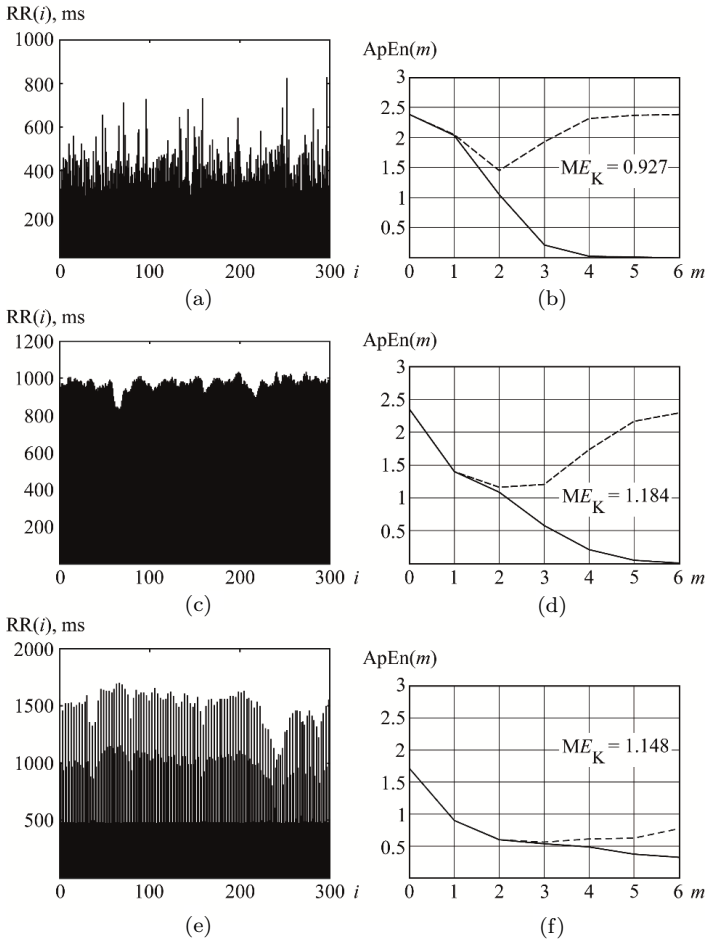


Figure 15.6. Examples of estimates ApEn(m) (solid line) and $\mathrm{ApEn}_{\mathrm{cor}}(m)$ (dashed line) for real signals of (a, b) atrial fibrillation; (c, d) normal rhythm; (e, f) frequent premature heartbeats.

Decision functions were constructed by estimates of approximate entropy using the theory of linear discriminant analysis [5]. The AF recognition problem involves the detection of fragments of this arrhythmia against the background of alternative rhythm groups. Therefore, the discriminant analysis was carried out for a two-class problem: AF (ω_1 class) and NR and FPHB (ω_2 class). The classification was conducted using a different set of parameters ApEn(m)

and $\mathrm{ApEn}_{\mathrm{cor}}(m)$. Informative features were selected by Fisher's criterion J. It made it possible to limit the description of objects to the following set of features: $\mathrm{M}E_{\mathrm{K}}; \mathrm{ApEn}(m)$ for $m = 1, 2, 3$. In this case, the theoretical classification errors were minimal and amounted to less than 0.1%. The effectiveness of the decision rule was verified using a control sample that includes 100 ECG realizations for each of the three types of rhythm. As shown by the results of calculations, this method has a high level of sensitivity ($S_e = 0.98$) and specificity ($S_p = 0.985$) in AF recognition. The obtained data (the ApEn method) were compared with AF detection results using alternative recognition methods [13, 30]:

- testing the Gaussian distribution law hypothesis (χ^2 criterion),
- estimation of the standard deviation of the rhythmogram (Dev),
- autoregressive model prediction (AR-model),
- estimation of the minimum of the refined conditional entropy (ME).

The results of the analysis of the effectiveness of these methods are given in Table 15.1.

The table shows that the AF detection method based on the approximate entropy is the most reliable.

15.3.4 *EEG-based anesthesia depth estimation*

The considered entropy characteristics were also used for the analysis of the electroencephalogram (EEG signal) in different stages of anesthesia [27, 40] and recognition of the stage of deep

Table 15.1. Effectiveness of recognition methods.

Method	AF detection against the background of	$S_e, \%$	$S_p, \%$
χ^2 criterion	NR	74.5	97.4
Dev	FPHB	98.8	64.7
AR model	NR and FPHB	93.3	91.4
ME	NR and FPHB	98.5	89.3
ApEn	NR and FPHB	98.0	98.5

anesthesia [28,38,39]. In [7], it was proposed to use the index ApEn(2) to estimate the depth of anesthesia by EEG.

The intensity of regular and chaotic changes in the EEG signal is different in different stages of anesthesia. A high level of brain activity is accompanied by a complex chaotic EEG signal. The decrease in the degree of functional activity is reflected in the EEG by regular low-frequency oscillations of sufficiently large amplitude. In Figure 15.7, four fragments of the EEG signal recorded from one patient at different stages of anesthesia are shown as an example: (a) before anesthesia, (b) in the initial stage, (c) during emergence from anesthesia, and also (d) in the final awakening stage. The sampling frequency of the EEG signal is 500 Hz. The fragment duration is 10 s. Figure 15.8 shows the result of calculating the approximate entropy for two consecutive 5-s fragments of these EEG signals with the value $r = 0.2\sigma_x$.

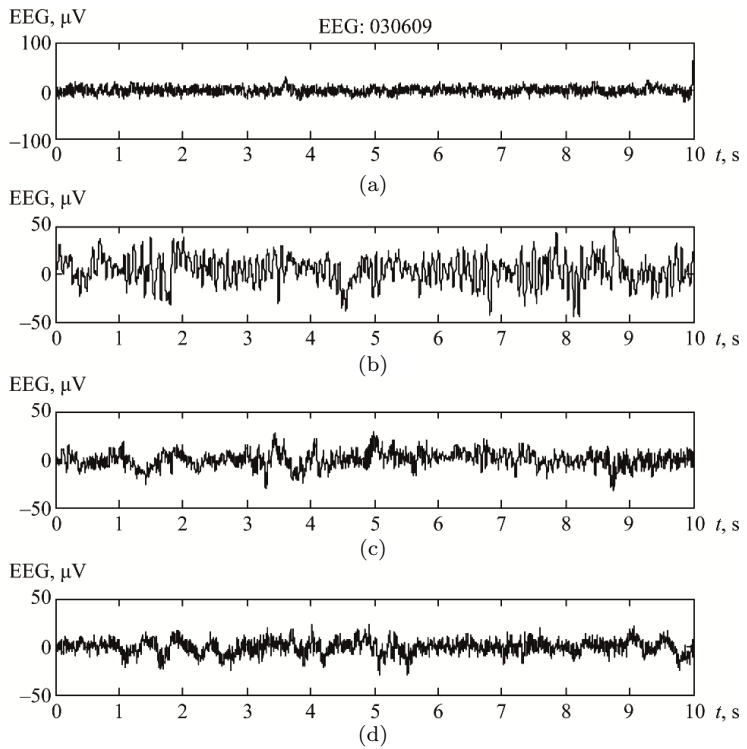


Figure 15.7. Fragments of the EEG signal in different stages of anesthesia: (a) before anesthesia; (b) deep anesthesia; (c) exit from anesthesia; (d) definitive awakening.

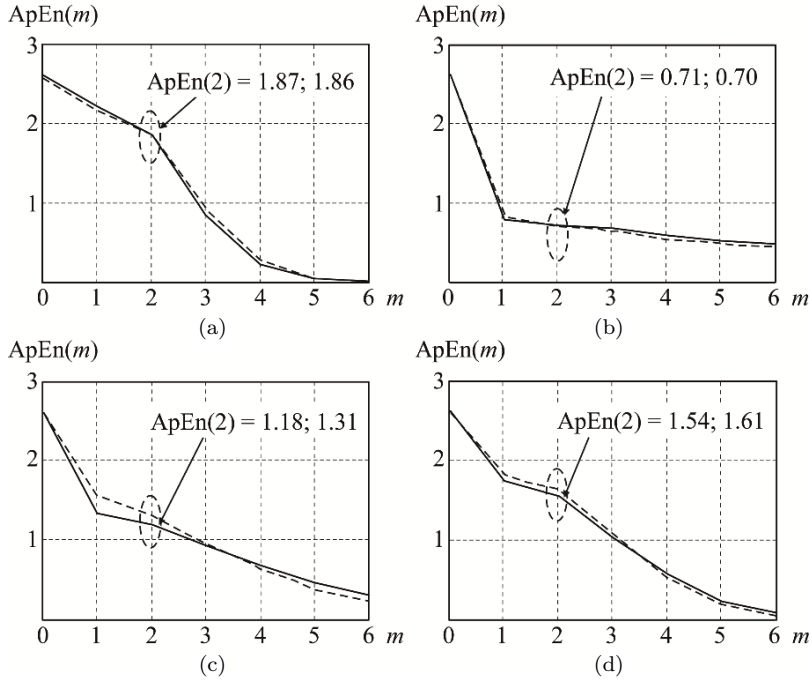


Figure 15.8. Approximate entropy ApEn(m) for EEG signals corresponding to different stages of anesthesia: (a) before anesthesia; (b) deep anesthesia; (c) emergence from anesthesia; (d) final awakening.

In the initial region of the change in parameter m, the curves have obvious differences at different functional states of the brain, as can be seen from the graphs of dependences ApEn(m). In this case, the values of parameter ApEn(2) reduce by more than twice when transitioning from the active state (Figure 15.8(a)) to the stage of deep anesthesia (Figure 15.8(b)). The emergence from anesthesia (Figure 15.8(c)) and the transition to the final awakening (Figure 15.8(d)) are accompanied by a gradual increase in the values of ApEn(2).

We used the normalization for the value of the absolute entropy ApEn(0) and experimentally selected the nonlinear scale of transition to final indices to bring the entropy index to the scale adopted in anesthesiology (0...100). In this case, entropy values close to 100 for the EEG signal recorded before the anesthesia starts and at the stage of full awakening should be expected. In the case of deep anesthesia, it is obvious that the index ApEn(2) will have small values.

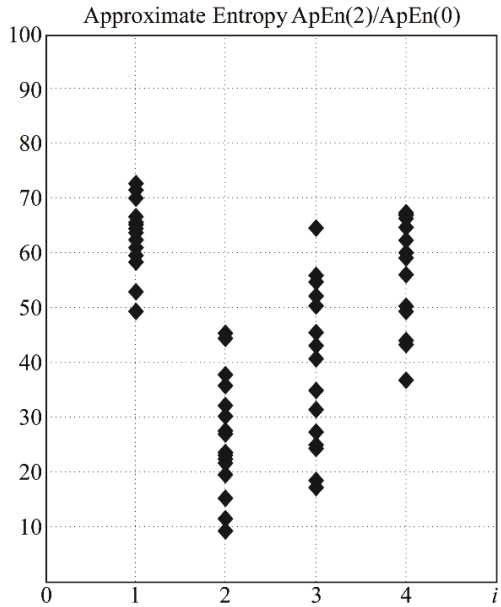


Figure 15.9. Mapping of entropy parameters along the directions i corresponding to the four stages of anesthesia.

Figure 15.9 shows the results of calculating the normalized index ApEn(2)/ApEn(0) in the accepted measurement scale for the sample of EEG signals (64 realizations, 5 s each) detected in eight patients in the clinical setting during the operation. They are represented graphically as a scatter of points on vertical lines with coordinates i (1) before the anesthesia begins, (2) deep anesthesia, (3) emergence from anesthesia, and (4) final awakening. As can be seen from the figure, disjoint sets of objects are formed on lines 1 and 2. Obviously, it is possible to solve the problem of recognizing the stage of deep anesthesia for the data sample under consideration with a threshold of 0.47. Stages 3 and 4 form groups of points that occupy an intermediate position between states 1 and 2. This corresponds to the existing concept of the process of emergence from anesthesia.

The Fisher's criterion J [5] was selected as an indicator of effectiveness of the use of the index ApEn(2)/ApEn(0) for recognition of the stages of anesthesia. It was calculated from the sets of values of the entropy parameter for two consecutive states ($1 \to 2; 2 \to 3; 3 \to 4$). In total, electroencephalograms of 31 patients were analyzed

during the experiment. The following values were obtained as a result of processing 124 EEG realizations of 10 s each: $J_{1,2} = 3.07$; $J_{2,3} = 0.78$; $J_{3,4} = 0.47$. Entropy parameters were estimated for short 5-second fragments of the EEG signal.

As follows from the analysis of the values of the J criterion, as well as the data shown in Figure 15.9, the method for estimating the parameter ApEn(2) makes it possible to solve the problem of recognizing two states: wakefulness and deep anesthesia. This provides a basis for further research on the entropic approach when recognizing intermediate anesthesia stages.

15.4 Detection of Dangerous Arrhythmias from the Electrocardiosignal Spectral Description

The problem of reliable detection of dangerous arrhythmias (flutter and fibrillation of heart ventricles) from ECG is a priority for cardiac monitoring systems. It can be solved on the basis of spectral analysis of the electrocardiogram [43] which becomes wavy in these disturbances and has a rather characteristic description in the frequency domain. It was shown [3] that the main electrocardiosignal (ECS) power is in the frequency range of 3–10 Hz during cardiac ventricular fibrillation. This differs sharply from the broadband ECS spectrum for a normal rhythm, the fundamental harmonics of which are in the frequency range of up to 25 Hz.

Analysis of the spectral description assumes the estimation of the power of the main harmonic components as well as the identification of some features of the distribution of the power spectral density (PSD) [32]. In problems of recognition of biomedical signals, this often leads to the selection of content features that are clearly different for various classes (for example, the presence of narrowband spectrum components and the intensity of the main PSD peak) [37].

It is possible to solve the problem of classifying signals in the investigation of the distribution of objects in the space of spectral features by implementing linear decision functions or constructing more complex decision surfaces. However, the optimization of the type of decision rules often does not lead to the required quality of signal recognition. This is due to the fact that the selected features, sometimes picked intuitively or by experience, are not significant for

the classification of information. Thus, the system of features can be incomplete, and the addition of some of them to the decision rules can even reduce the effectiveness of the recognition system. Indeed, the selection of a set of features that provide the necessary classification quality is a difficult task [46]. In [14, 15, 21], it was proposed to solve it by making a classification description that considers the complete series of spectral coefficients as an ordered set of parameters. This leads to the possibility of constructing an effective separating hyperplane in the space of selected features [16, 17, 22].

15.4.1 *Selecting the frequency band for analyzing the PSD function*

Let the original signal $x(t)$ be a sequence of N samples x_n, $n = 0, \ldots, (N-1)$ given with a sampling rate of $f_d = 1/h$ Hz. Then a selective PSD estimate can be obtained on the time interval $T = Nh$ [32]:

$$G(f_k) = \frac{2h}{N} \left| \sum_{n=0}^{N-1} x_n \exp\left(-j\frac{2\pi kn}{N}\right) \right|^2,$$

where $f_k = k/T = k/Nh; k = 0, 1, \ldots, N-1$.

Next, we analyze the obtained PSD estimate in the range of values of $k = 0, 1, 2, \ldots, N/2$, which corresponds to the spectrum limitation to $f_c = f_d/2$ (Nyquist frequency).

Considering the known effect of energy leakage through the side lobes of the spectral window, which results from the truncation of the process under investigation by a rectangular window with a width of T, it is necessary to first smooth the ends of the initial implementation. Special weighting functions, for example, cosine one, are used for this purpose. They make it possible to reduce the bias of the sample estimate of the spectrum [32].

The obtained PSD estimate can be used to generate spectral features and construct decisive rules. However, it is necessary first to limit the analyzed frequency range, which will reduce the dimension of the original spectral description and simplify the signal recognition procedure. To do this, we can estimate the discrepancy between the selected signal descriptions by generating training data samples for given classes (ω_1 and ω_2) and calculating

PSD estimates for them. The degree of closeness between the given signal classes is estimated by arranging sequentially descriptions $\mathbf{G}^{(k)} = (G(f_1), G(f_2), \ldots G(f_k)), k = 1, 2, \ldots$ obtained by selecting k first spectral components in the PSD. A criterion based on the analysis of the mean intergroup distance can be used as a measure of closeness between ω_1 and ω_2:

$$\rho^{(k)}(\omega_1, \omega_2) = (1/n_1 n_2) \cdot \sum_{\mathbf{G}_q^{(k)} \in \omega_1} \sum_{\mathbf{G}_j^{(k)} \in \omega_2} \rho(\mathbf{G}_q^{(k)}, \mathbf{G}_j^{(k)}), \qquad (15.8)$$

where $\mathbf{G}_q^{(k)}, \mathbf{G}_j^{(k)}$ is the spectral representation of the objects q and j from the classes ω_1 and ω_2 used at the kth step of the computational procedure (for k frequency components), $\rho(\mathbf{G}_q^{(k)}, \mathbf{G}_j^{(k)})$ is the Euclidean distance between the objects q and j in the k-dimensional feature space, and n_1 and n_2 are the numbers of objects constituting classes ω_1 and ω_2.

The number of the first spectral features that make the greatest contribution to the estimation of the difference between classes can be determined by constructing the dependence of the quantity $\rho^{(k)}(\omega_1, \omega_2)$ on k.

This approach was used in ECS processing for solving the problem of recognizing a dangerous rhythm disturbance, ventricular fibrillation (VF) (ω_1 class), against the background of atrial and ventricular dysfunctions that do not pose a threat to the life of the patient, background rhythm (BR) (class ω_2). In particular, the BR class included a normal rhythm with single premature beats, group premature beats, and other arrhythmias against the background of which VF developed [14, 21, 22].

As an example, Figure 15.10 shows ECS implementations and corresponding selective estimates of the normalized spectrum for (a, b) ventricular fibrillation (VF) and (c, d) background rhythm (BR). The signal fragments are represented by 2-s ECS implementations sampled at a frequency of 250 Hz (50 units of ADC correspond to 1 mV).

Analysis of the obtained dependences made it possible to isolate the effective area of PSD analysis with an upper frequency of $f_{\max} = 14$ Hz ($k = 28$), since higher frequencies, as can be seen from the figures, do not contribute significantly to the estimated discrepancy between the classes.

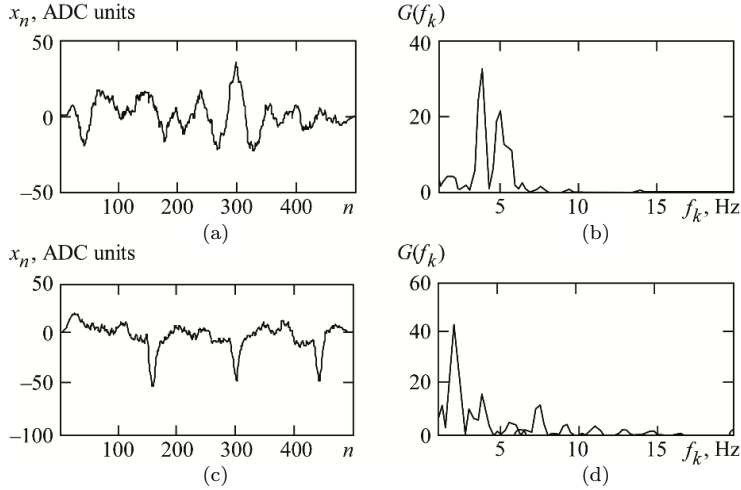


Figure 15.10. ECS realizations and the corresponding sample estimates of the normalized spectrum: (a, b) for ventricular fibrillation; (c, d) for the background rhythm.

15.4.2 *Generation of a set of spectral features*

Next, we group the spectral coefficients by smoothing their values within the boundaries of the given ranges. It is known that the sample spectrum calculated for independent frequencies f_k obeys the law of χ^2-distribution with two degrees of freedom. The value obtained for each frequency is not a consistent estimate of the PSD because its standard deviation is comparable in magnitude to the mean value of $G(f_k)$ [32]. We proceed to χ^2-distribution with $2l$ degrees of freedom, which will lead to a decrease in the magnitude of the random error by a factor of $\sqrt{l}$, using smoothing with respect to l independent frequencies. Thus, the procedure of combining adjacent frequencies helps to reduce the dispersion of the PSD estimate, i.e. leads to more stable spectral features.

A set of spectral features can be generated by one of the proposed methods. The first method is based on the *Daniell periodogram* estimate that generates a new series of spectral samples by averaging the $(2P + 1)$ neighboring frequencies:

$$G_{\mathrm{D}}(f_i) = \frac{1}{2P+1} \sum_{n=i-P}^{i+P} G(f_n).$$

The second method involves automatic generation of effective frequency domains by finding optimal boundaries for grouping adjacent spectral coefficients. First, the maximum allowed number of intervals L is set. Point PSD estimates are combined within each of them. This value also determines the number of features that can be used in constructing discriminant functions. Then, the corresponding areas under the PSD curve are estimated for each pth feature:

$$G_{Lp} = \frac{1}{Nh} \sum_{k=m_p}^{m_{p+1}} G(f_k), p = 1, \ldots, L,$$

where L is the number of spectral features (the number of given frequency areas) and m_p and m_{p+1} are the lower and upper bounds of the frequency domain $(f_{m_p}, f_{m_{p+1}})$ used to calculate the pth feature.

The search for the location of the bounds (m_p, m_{p+1}) in each of the L frequency domains makes it possible to obtain a sufficiently large set of vectors $\{\mathbf{G}^{(L)}\}$, from which the best one can be determined by the maximum value of criterion (15.8). The gradient search procedure for the extremum of the objective function can be used to reduce the analysis time.

The following frequency domain bounds were obtained as a result of applying this processing method to the sample of real ECG records:

$$(0.244; 2.44; 5.124; 10.004; 14.64) \text{ Hz} \quad \text{for} \quad L = 4,$$

$$(0.244; 2.44; 5.124; 10.004; 12.444; 14.64) \text{ Hz} \quad \text{for} \quad L = 5.$$

All frequencies are multiple of 0.244 Hz, which is associated with the addition of zeros to the original sequence $x(n), n = 0, K, N-1$ up to $N = 1024$ samples.

15.4.3 *Construction of decision functions*

The final stage of the research is related to the recognition of the ECS classes ω_1 and ω_2. We used the well-known method of finding Fisher's linear discriminant to construct the decision functions in the space of spectral features [5]. A comparative analysis of the obtained spectral descriptions was carried out by the J optimization criterion.

Figure 15.11 shows the histogram distributions of the projections of objects onto the vector $\mathbf{W}(y = \mathbf{W}^{\mathrm{T}}\mathbf{G})$ for smoothed

PSD estimates in the frequency range $0 \ldots 15$ Hz. On the left (Figure 15.11(a)), the result obtained using the Daniell periodogram for $L = 15$ frequency samples with the window width $\Delta f = 0.976$ Hz is shown. On the right (Figure 15.11(b)), the distributions obtained during the generation of $L = 4$ frequency domains of the spectrum by the criterion of maximization of intergroup distances are presented. Here, the ω_1 class is represented by VF realizations and ω_2, by the BR signals.

The analysis of the obtained data shows that in both cases objects form quite compact groups, and the signals can be effectively classified using one of the proposed methods for constructing the space of spectral features. It is important to note that it is possible to significantly reduce the number of used features by optimizing frequency domains. As a result, the task of detecting VF was solved by generating four PSD frequency domains.

The results of testing the VF detection effectiveness on a control data sample that included more than 200 short ECG realizations from the MIT-BIH [34] database showed that the sensitivity of the method was $S_e = 0.98$ and the specificity, $S_p = 0.96$ [21].

A method for correcting the position of the discriminant function by finding an additional vector by Fisher's criterion J and transitioning to the 2D feature space of the vectors $\mathbf{W}_1$, $\mathbf{W}_2$ was considered in [22]. In the course of the experiments, it was found that the introduction of an additional vector $\mathbf{W}_2$ for the spectral description of the ECS using the Daniell periodogram estimate $(L = 15, \Delta f = 0.976, \text{Hz})$ increased the distance between the closest neighbors of the considered classes ω_1 (VF), ω_2 (BR) by 1.5 times.

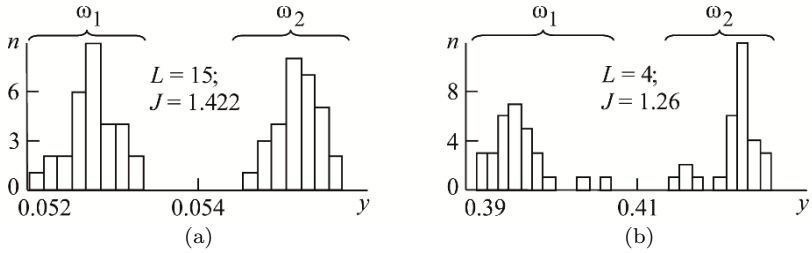


Figure 15.11. Distribution histogram of the projections of objects onto the vector $\mathbf{W}$ for the classes ω_1 and ω_2: (a) the Daniell window with $\Delta f = 0.976$ Hz was used; (b) frequency domains were optimized.

Figure 15.12 shows the results of mapping these classes of objects onto the space of new coordinates y_1, y_2 as an example (the corresponding histogram of the distribution of objects in the $\mathbf{W}_1$ direction is shown in Figure 15.11(a)).

As can be seen from the figure, the distance between the nearest neighbors of the classes under consideration increased from $r_1 = 0.0014$ to $r_2 = 0.0021$. It made it possible to correct the position of the separating hyperplane by transitioning from the linear function H_1 to H_2.

15.5 Discriminant Analysis of a Multiclass Problem using the Fisher's Criterion

In the practice of medical research, there are problems that require recognition of several (more than two) classes of biomedical signals.

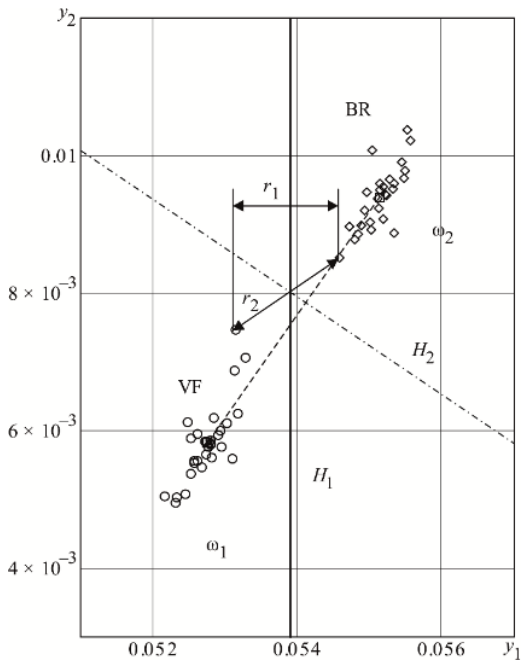


Figure 15.12. Construction of a decision function on the plane of two vectors $W1$, $W2$.

These include the detection of dangerous arrhythmias [33] and the classification of various rhythm disturbances during continuous monitoring of the patient's ECG [36]. In cardiac monitoring systems, it is necessary to reliably detect dangerous arrhythmias at the time of their sudden onset [8, 47], but it is equally important to recognize abnormalities that are their precursors. In addition, a reliable assessment of the patient's current condition is important to prevent the development of dangerous situations, which is related to the recognition of arrhythmias that do not pose a direct threat to the patient's life [6,47]. In this case, the problem of constructing decision functions for many classes of electrocardiosignals arises.

One of the ways to solve such a multiclass problem is a multiple discriminant analysis based on Fisher's criterion [5].

15.5.1 *Method of solving a multiclass problem based on the Fisher's criterion*

If there are c classes, it is necessary to construct a (c–1) decision function. This problem can be solved by reducing it to a set of two-class problems or based on multiple discriminant analysis. The latter approach uses Fisher's criterion J and the generalization of definitions for scatter matrices between classes and within classes. The J criterion, which estimates the degree of separation of the given classes of signals, is generally given by the trace of the matrix:

$$J = \mathrm{tr}(\mathbf{S}_W^{-1}\mathbf{S}_B), \tag{15.9}$$

where $\mathbf{S}_B$ is the scatter matrix between classes and $\mathbf{S}_W$ is the generalized scatter matrix inside the classes.

This criterion should be *maximized.*

Finding a linear transformation $\mathbf{Y} = \mathbf{W}^{\mathrm{T}}\mathbf{X}$ that maximizes the value of J for the case of two classes is reduced to determining only one eigenvalue of the matrix $(\mathbf{S}_W^{-1} \cdot \mathbf{S}_B)$. Hence follows the possibility of determining the matrix $\mathbf{W}$ with the maximum ratio of scattering between the classes to scattering within the classes in the following form [5]:

$$\mathbf{W} = \mathbf{S}_W^{-1}(\mathbf{M}_1 - \mathbf{M}_2),$$

where $\mathbf{M}_1$ and $\mathbf{M}_2$ are the vectors of the mean values of two classes.

In the case of c classes, in the transition from the L-dimensional feature space to the $(c-1)$-dimensional space object projections are found using the same matrix transformation $\mathbf{Y} = \mathbf{W}^{\mathrm{T}}\mathbf{X}$ but $\mathbf{W}$ is already a matrix with the size of $L \times (c-1)$. This matrix can be found if the matrix $\mathbf{S}_B$ in (15.9) is represented in the following form:

$$\mathbf{S}_B = \sum_{i=1}^{c} p_i(\mathbf{M}_i - \mathbf{M})(\mathbf{M}_i - \mathbf{M})^{\mathrm{T}},$$

where $p_i = n/n_i$ is the relative number of objects constituting the ith class (the frequency of their occurrence) and $\mathbf{M}$ is the general mean vector for all c classes.

The matrix $\mathbf{S}_W$ is defined as follows:

$$\mathbf{S}_W = \sum_{j=1}^{c} p_j \mathbf{S}_W^{(j)},$$

where $\mathbf{S}_W^{(j)}$ is the scatter matrix for the jth class (the sample covariance matrix).

The rank of the matrix $\mathbf{S}_B$ is $(c-1)$, and this value determines the number of eigenvalues and eigenvectors defined by the matrix $\mathbf{W}$.

Finding the elements of the matrix $\mathbf{W}$ is reduced to determining the eigenvalues of the matrix $(\mathbf{S}_W^{-1} \cdot \mathbf{S}_B)$. They can be found as the roots of the characteristic equation $|\mathbf{S}_B - \lambda_i \mathbf{S}_W| = 0$, and then $d = c - 1$ eigenvectors $\mathbf{W}_i$ are obtained as a solution of the system of equations $(\mathbf{S}_B - \lambda_i \mathbf{S}_W) \cdot \mathbf{W}_i = 0$. In a new space with dimension d, the J criterion is defined as $J = \sum_{i=1}^{d} \lambda_i$.

Eigenvectors with nonzero eigenvalues define the $(c-1)$-dimensional space in which discriminant functions can be constructed and decision rules can be defined.

15.5.2 *Introduction of weight functions in the J optimization criterion*

As shown in [9], the J criterion becomes an indicator of large intergroup distances and weakly reflects the relative location of closely located classes as the number of classes increases. The procedure for constructing decision rules can be optimized by reducing it to a set

of pairwise classification problems with the introduction of weight coefficients $a_{i,j}$, which increase the effect of closely located classes on the J criterion. In this case, the matrix $\mathbf{S}_B$ is calculated as follows:

$$\mathbf{S}_B = \sum_{i=1}^{c-1} \sum_{j=i+1}^{c} p_i p_j (\mathbf{M}_i - \mathbf{M}_j)(\mathbf{M}_i - \mathbf{M}_j)^{\mathrm{T}} = \sum_{i=1}^{c-1} \sum_{j=i+1}^{c} p_i p_j \mathbf{S}_B^{(i,j)},$$

and the generalized expression for the J criterion takes the following form:

$$J = \sum_{i=1}^{c-1} \sum_{j=i+1}^{c} p_i p_j a_{i,j} \mathrm{tr}[(\mathbf{W}^{\mathrm{T}} \mathbf{S}_W \mathbf{W})^{-1} \cdot (\mathbf{W}^{\mathrm{T}} \mathbf{S}_B^{(i,j)} \mathbf{W})], \quad (15.10)$$

where $p_i = n_i/n$ and $p_j = n_j/n$ are the frequencies of occurrence of objects that constitute classes ω_i and ω_j.

In the cited paper, it was proposed to use weights in the form of an error function:

$$\mathrm{erf}(\eta - t/\sigma),$$

where t is the boundary of the decision rule and η and σ are the distribution parameters. This function is calculated for given groups of objects (i and j) based on assumptions about the normal distribution law and the equality of covariance matrices. This approach seems to be effective since the J criterion can be approximated to the estimate of the reliability of object recognition by summing the probabilities of the correct solution for pairwise classification.

In this case, the weight coefficients $a_{i,j}$ are defined as follows:

$$a_{i,j} = \frac{1}{2(\Delta_{i,j})^2} \cdot \mathrm{erf}\left(\frac{\Delta_{i,j}}{2\sqrt{2}}\right),$$

where $\Delta_{i,j}$ is the Euclidean distance between the centers of the classes ω_i and ω_j.

In the region of small values of $\Delta_{i,j}$, for example, when objects are recognized by the parameters of a normalized spectrum or other normalized parameters, it is possible to specify the weights $a_{i,j}$ using the error function approximation by polynomial function [36]:

$$a_{i,j} \approx \frac{1}{8\sqrt{\pi} x_{i,j}} \left(1 - \frac{x_{i,j}^2}{3} + \frac{x_{i,j}^4}{2!\,5}\right),$$

where $x_{i,j} = \left(\Delta_{i,j}/2\sqrt{2}\right), \Delta_{i,j} \leq \sqrt{2}$.

The determination of the elements of the matrix $\mathbf{W}$ is reduced to the problem of determining the eigenvalues of the matrix:

$$\mathbf{S}_W^{-1} \cdot \sum_{i=1}^{c-1} \sum_{j=i+1}^{c} p_i p_j a_{i,j} \cdot \mathbf{S}_B^{(i,j)}.$$

Experiments showed [18, 23] that this method is effective in the case of an uneven mutual arrangement of the centers of classes, i.e. when the values of $\Delta_{i,j}$ for different classes are significantly different.

15.5.3 *Application of the method for detecting dangerous arrhythmias*

The considered methods for solving multiclass problems were used in the detection of rhythm disturbances representing different degrees of danger to the patient's life [19,24–26]. In particular, the problem of detection of three classes of arrhythmias was solved: ω_1 indicates ventricular fibrillation (VF) and ventricular flutter (VFLU), ω_2 refers to paroxysmal ventricular tachycardia (RVT) and background rhythm (BR) represented by different forms of premature heartbeats, and ω_3 is torsades de pointes (TdP). Experimental studies were aimed at creating reliable algorithms for ECG detection of extremely dangerous disorders, VF and VFLU, but at the same time, the possibility of detecting these arrhythmias at early stages was analyzed, namely, at the time of RVT transformation into VFLU, which is usually accompanied by the short period of TdP.

The effectiveness of the application of the considered procedures was evaluated based on the results of experiments performed on real data including ECS records lasting more than 20 min. All ECG realizations were derived from the standard MIT-BIH computer database of ECG signals [34]. The experimental material used in constructing the decision functions included 150 2-s ECG fragments. Short fragments were selected due to the need for urgent measures to restore the contractile function of the heart in online mode.

Signals were recognized in the frequency domain, which is associated with the high informativeness of the ECS spectral description for the analysis of various forms of dangerous arrhythmias [2, 4]. An ordered set of 28 spectral features obtained in the frequency range of

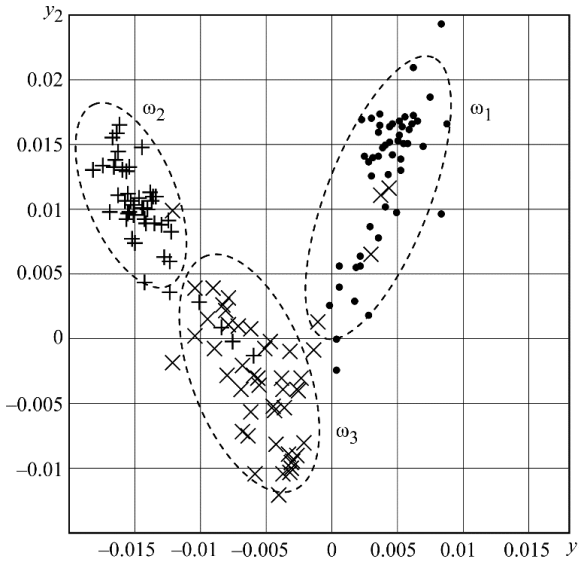


Figure 15.13. The representation of objects of three classes ω_1 (VF and VFLU), ω_2 (RVT and BR), and ω_3 (TdP) in the space of vectors $\mathbf{W}_1, \mathbf{W}_2$ obtained using weight functions.

0–15 Hz using overlapping segments was used as an initial description of the objects of each class.

Figure 15.13 shows the mapping of objects of three classes of signals $\omega_1, \omega_2, \omega_3$ onto the plane (y_1, y_2) of eigenvectors $\mathbf{W}_1, \mathbf{W}_2$. 2D space was optimized by the J criterion using weight functions.

In the course of the experiments, discriminant functions were constructed, the boundaries of the feasible regions were determined, and classification errors were found. A comparative analysis of the obtained 2D mappings showed [25,26] that the introduction of weight functions leads to a more uniform grouping of objects on the plane. At the same time, the mean classification error can be reduced from 8.2% to 4.6% which indicates the effectiveness of this optimization procedure.

Acknowledgments

The study was supported by a grant from the Russian Science Foundation No. 23-21-00215, https://rscf.ru/project/23-21-00215/.

References

[1] Akay, M. (ed.) (2001) Nonlinear Biomedical Signal Processing, Vol. 2 Dynamic Analysis and Modelling (IEEE, New York).

[2] Clayton, R.H., Campbell, R.W.F., and Murray, A. (1997) Time–frequency analysis of human polymorphic ventricular tachycardia. *Computing in Cardiology*, 24, 97–100.

[3] Clayton, R.H. and Murray, A. (1998) Comparison of techniques for time-frequency analysis of the ECG during human ventricular fibrillation. *IEEE Proceedings on Science, Measurement and Technology,* 145, 301–306.

[4] Clayton, R.H., Murray, A., and Campbell, R.W.F. (1994) Frequency analysis of self–terminating ventricular fibrillation (IEEE Computer Society Press), pp. 705–708.

[5] Duda, R.O., Hart, P.E., and Stork, D.G. (2001) *Pattern Classification, Part 1* (Wiley, New York).

[6] Ilankumaran, V. and ThamaraiSelvi, S. (2011) Ventricular arrhythmias detection using wavelet decomposition. *International Journal of Computer Applications*, 1, 11–18.

[7] Kalinichenko, A. N., Manilo, L.A., and Nemirko, A.P. (2015) Analysis of anesthesia stages based on the EEG entropy estimation. *Pattern Recognition and Image Analysis,* 4, 632–641.

[8] Kaur, L. and Singh, V. (2013) Ventricular fibrillation detection using empirical mode decomposition and approximate entropy. *International Journal of Emerging Technology and Advanced Engineering,* 5, 260–268.

[9] Loog, M., Duin, R.P.W., and Haeb-Umbach, R. (2001) Multiclass linear dimension reduction by weighted pairwise Fisher criteria. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 23, 762–766.

[10] Manilo, L. A. (2005) Identification of biosignals on application estimations conditional entropy. *Izvestiya SPbGETU "LETI" Series of "Biotechnical Systems in Medicine and Ecology",* 2, 53–59 (in Russian).

[11] Manilo, L.A. (2008) Detection of biological signals with chaotic properties through assessment of conventional entropy. Proceedings of the 9th International Conference "Pattern Recognition and Image Analysis: New Information Technologies" (PRIA-9-2008) (Nizhni Novgorod), Vol. 2, pp. 11–14.

[12] Manilo, L.A. (2009) Atrial fibrillation recognition in cardiological systems for diagnostics and monitoring. *Biotechnosphere*, 2, 41–45 (in Russian).

[13] Manilo, L.A. (2004) Autoregressive models of random process in recognition of the cardiac rhythm disturbances. *Izvestiya SPbGETU "LETI" Series of "Biotechnical Systems in Medicine and Ecology",* 2, 3–8 (in Russian).

[14] Manilo, L.A. (2006) Ordering of spectral attributes by empirical estimations of intergroup distance in problems of biosignals classification. *Proceedings of the Russian Universities: Radioelectronics,* 3, 20–29 (in Russian).

[15] Manilo, L.A. (2003) Measurement and computerized analysis of signals in biomedical systems. *Proceedings of the V Intern. Symposium on Test and Measurement*, China, June 1–5, Vol. 1, pp. 867–870.

[16] Manilo, L.A. (2005) Fisher linear discriminant analysis in pattern recognition of biosignals by frequency properties. *Proceedings of 12th Russian Conference on Mathematical Methods for Pattern Recognition, MMPR-12*, Moscow, pp. 371–374 (in Russian).

[17] Manilo, L.A. (2006) Construction of discriminant functions for cardiological studies in frequency domain. *Izvestiya SPbGETU "LETI" Series of "Biotechnical Systems in Medicine and Ecology",* 1, 13–21 (in Russian).

[18] Manilo, L.A. (2006) Construction of discriminant functions for cardiological studies in frequency domain. *Izvestiya SPbGETU "LETI" Series of "Biotechnical Systems in Medicine and Ecology",* 1, 13–21 (in Russian).

[19] Manilo, L.A. (2016) Multiple discriminant analysis for recognition of electrocardiosignals in frequency domain. *Biomedical Radioelectronics*, 8, 9–13 (in Russian).

[20] Manilo, L.A. and Nemirko, A.P. (2009) Approximation of the Kolmogorov entropy in the analysis of chaotic processes on finite samples. *Proceedings of 14th Russian Conference on Mathematical Methods for Pattern Recognition, MMPR-14*, Suzdal, pp. 405–407 (in Russian).

[21] Manilo, L.A. and Nemirko, A.P. (2005) Forming the spectral signs when classifying the electrocardiosignals within the frequency range. *Pattern Recognition and Image Analysis,* 4, 668–671.

[22] Manilo, L.A. and Nemirko, A.P. (2016) Recognition of biomedical signals based on their spectral description data analysis. *Pattern Recognition and Image Analysis*, 4, 782–788.

[23] Manilo, L.A. and Nemirko, A.P. (2007) Reduction of the dimensionality of the spectral features space in the multi-class problem of signal recognition. *Proceedings of 13th Russian Conference on Mathematical Methods for Pattern Recognition, MMPR-13*, Zelenogorsk, pp. 356–359 (in Russian).

[24] Manilo, L. and Nemirko, A. (2016) Multiple discriminant analysis for recognition of biosignals in frequency domain. *Intelligent Data Processing: Theory and Applications, Abstracts of the 11 International Conference IDP-2016*, Barcelona, p. 143.

[25] Manilo, L.A. and Nemirko, A.P. (2016) Reduction of the spectral signs space dimension in the multiclass task of the ECG-signals distinguishing. *Pattern Recognition and Information Processing, Proceedings of the 13th International Conference PRIP'2016*, Minsk, pp. 156–158.

[26] Manilo, L. and Nemirko, A. (2017) Reduction of the spectral feature space dimension in the multiclass problem of ECG-Signals recognition. In Krasnoproshin, V.V. and Ablameyko, S.V. (eds.) *Pattern Recognition and Information Processing, PRIP 2016, CCIS*, Vol. 673 (Springer International Publishing AG), pp. 114–118.

[27] Manilo, L.A. and Volkova, S.S. (2012) Approximate entropy parameters analysis in the context of anesthesia depth Estimation. *Biomedical Radioelectronics,* 1, 58–61 (in Russian).

[28] Manilo, L.A. and Volkova, S.S. (2013) Recognition of the deep anesthesia stage from parameters of the approximated entropy of EEG signal. *Pattern Recognition and Image Analysis,* 1, 92–97.

[29] Manilo, L.A. and Zozulya, E.P. (2007) Research of the opportunity of application approximate entropy for the analysis of biosignals. *Izvestiya SPbGETU "LETI" Series of "Biotechnical Systems in Medicine and Ecology",* 1, 3–9 (in Russian).

[30] Manilo, L.A. and Zozulya, E.P. (2006) Automatic recognition of atrial fibrillation via estimation of approximated entropy. *Informatsionno-upravliaiushchie sistemy (Information and Control Systems),* 1, 21–27 (in Russian).

[31] Manilo, L.A. and Zozulya, E.P. (2009) Estimation of time series correlation dimension for the task of cardiac rhythm disturbances analysis. *Biomedical Radioelectronics*, 11, 32–39 (in Russian).

[32] Marple, S.L. (1987) *Digital Spectral Analysis with Applications* (Prentice Hall, Englewood Cliffs).

[33] Meng, Q., Xia, D., Zhang, Q., and Zhang, Z. (2016) Detection ventricular tachycardia and fibrillation using the Lempel-Ziv complexity and wavelet transform. *WSEAS Transactions on Information Science and Applications,* 13, 118–125.

[34] MIT-BIH Arrhythmia Database. Massachusetts Institute of Technology. http://www.physionet.org/physiobank/database/mitdb; http://ecg.mit.edu.

[35] Moon, F.C. (2004) Chaotic Vibrations: An Introduction for Applied Scientists and Engineers (John Wiley & Sons, Hoboken).

[36] Moraes, J.C.T.B., Blechner, M., Vilani, F.N., and Costa, E.V. (2002) Ventricular fibrillation detection using a leakage/complexity measure method. *Computers in Cardiology,* 29, 213–216.

[37] Nemirko, A.P., Manilo, L.A., and Degtyareva, I.N. (2001) Recognition of heart ventricular fibrillation on the basis of analysis of the electrocardiogram in the frequency domain. *Pattern Recognition and Image Analysis,* 2, 353–355.

[38] Nemirko, A.P., Manilo, L.A., Kalinichenko, A.N., and Volkova, S.S. (2010) Use of EEG signal entropy indicators for pattern recognition of anesthesia stages. *Intelligent Data Processing, Proceedings of the 8 International Conference IDP-2010*, Cyprus, Paphos, pp. 454–457 (in Russian).

[39] Nemirko, A.P., Manilo, L.A., Kalinichenko, A.N., and Volkova, S.S. (2010) Entropy methods of anesthesia depth estimation based on EEG analysis. *Informatsionno-upravliaiushchie sistemy (Information and Control Systems),* 3, 69–74 (in Russian).

[40] Nemirko, A.P., Manilo, L.A., Kalinichenko, A.N., and Volkova, S.S. (2010) Comparison of various entropy-based techniques of EEG analysis for the anesthesia stages recognition. *Biotechnosphere*, 3, 3–10 (in Russian).

[41] Pincus, S.M. (1991) Approximate entropy as a measure of system complexity. *Proceedings of the National Academy of Sciences USA,* 88, 2297–2301.

[42] Porta, A., Baselli, G., Liberati, D., Montano, N., Cogliati, C., Gnecchi-Ruscone, T., Malliani, A., and Cerutti, S. (1998) Measuring regularity by means of a corrected conditional entropy in sympathetic outflow. *Biological Cybernetics,* 78, 71–78.

[43] Rangayyan, R.M. (2002) *Biomedical Signal Analysis: A Case-Study Approach* (John Wiley & Sons Inc.).

[44] Schuster, H.G. (1988) *Deterministic Chaos – An Introduction*, 2nd Ed. (VCH Publishers, Weinheim).

[45] Shannon, C.E. (1948) A mathematical theory of communication. *Bell System Technical Journal*, 3, 379–423; 4, 623–656.

[46] Tou, J.T. and Gonsalez, R.C. (1974) *Pattern Recognition Principles* (Addison-Wesley Publishing Company).

[47] Verma, A. and Dong, X. (2016) Detection of ventricular fibrillation using random forest classifier. *Journal of Biomedical Science and Engineering*, 9, 259–268.

© 2025 World Scientific Publishing Company
https://doi.org/10.1142/9789811267215_0016

Chapter 16

Brain Signal Analysis from the Standpoint of the Structural Approach

Yakov A. Furman*, Victor V. Sevastyanov, and Konstantin O. Ivanov†

Federal State Budgetary Educational Institution of Higher Education "Volga State University of Technology", Yoshkar-Ola, Mari El Republic, the Russian Federation

**FurmanYA@volgatech.net*

†*IvanovKO@volgatech.net*

In this chapter, elements of mathematical formalism of subjective modeling of categories of fuzziness and uncertainty that reflect incompleteness and unreliability of information used in construction of a subjective morphological model are used and mathematical foundations of morphological image analysis, oblique image projection formalism, analysis of dependency of relative image shapes, and connectedness of their absolute shapes are considered. A numerical measure of rival similarity is defined on a set of image shapes. It measures a degree of similarity of an image shape to another shape with respect to the third one.

Examples of subjective morphological models and optimization of decisions based on them using oblique projection and subjective optimization are given. It is shown that optimal solutions to morphological problems of image analysis and interpretation are formulated in terms of oblique image projection and subjective optimization.

Keywords: Electroencephalogram, primitive element, structural (syntactic) approach, EGG segmentation, EEG computer processing.

16.1 Introduction

The human brain generates electrical potential oscillations as quasi-stationary rhythms of various frequencies. Recorded on the head surface, these potentials, the electroencephalogram (EEG), serve as an objective indicator of brain activity and make the latter explorable in any situation: during wakefulness, sleep, normal daily activities, or under pathological conditions up to complete loss of consciousness. Research into the dynamics of these potentials underpins the understanding of human brain activity as a complex system [16].

The origin of the scientific approach to studying brain potentials dates back to 1928, when H. Berger, an Austrian doctor, discovered the rhythmic electrical brain activity with a frequency of 10 Hz (α-rhythm) recorded on a patient's intact scalp [16]. In the middle of the last century, delta and beta rhythms, observed in other EEG frequency bands, were discovered, as well as the relations between the rhythms and pathological brain states. The research achievements led to the development of clinical EEG based on the visual inspection of waveforms. Mathematical methods of EEG analysis, that were mainly related to the EEG spectra and correlation functions, gained impetus for development.

In recent years, neuroimaging methods have become increasingly common in neurology, among them are computed tomography (CT), magnetic resonance imaging (MRI), positron emission tomography (PET), etc. These methods provided a solution to the problem of differentiating the affected areas of the brain by visualizing its internal structures. Thus, both neuroimaging and EEG are effective and objective methods for studying the same object: the human brain. The critical distinction between these methods is as follows:

Clinical EEG enables diagnosis of the functional activity of the central nervous system (CNS), i.e. the key characteristic of a living human brain in its normal state and in the event of neurological disorders. The neurovisualization methods allow for successful diagnosis of organic brain lesions [18]. The main purposes of EEG include research into "functional" abnormalities: neurological,

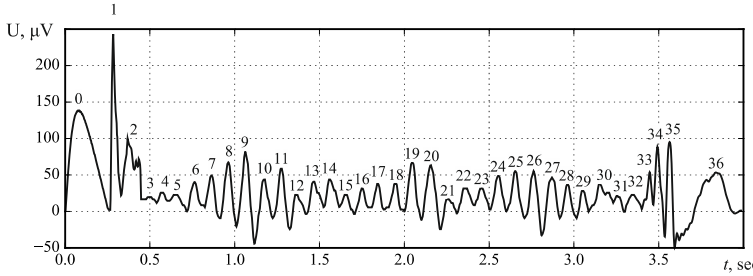


Figure 16.1. An EEG epoch containing heterogeneous patterns.

mental, emotional, behavioral, and cognitive disorders, as well as psychosomatic diseases. Subsequent studies, e.g. those on evoked potentials, have revealed that the EEG is not only capable of reflecting the overall functional state of the brain, but it also responds to the environment, i.e. the structure of visual and auditory images, signals from upper limbs [13].

An EEG fragment containing potential oscillations in the 0–50 Hz band is presented in Figure 16.1.

Since the early days of EEG until the 1950s, clinical conclusions were made by experienced clinicians who described the waveform shapes verbally, based on the outcome of visual inspection. Visual analysis is quite a labor-intensive technique that yields subjective results due to utilizing the doctor's ability to remember the waveforms, estimate them visually, and compare and recognize their shapes. Clinical evaluation comprises two parts: the first one is purely descriptive, while the second ("opinion", or "diagnostic conclusions") involves diagnostic assessment of the EEG record. The doctor starts "reading" an EEG with a cursory review of the records to search for activity areas exhibiting deviations from the norm. Next, the doctor describes the different activity types detected and only after that proceeds to the diagnostic assessment. The basic EEG element is the "wave", i.e. a piece of record between two adjacent extrema of the same sig [2].

Methods of EEG computer analysis (computer-analyzed EEG, CEEG) are free from the drawbacks of visual analysis. The term 'CEEG' means a set of methods and techniques that enable the analysis of recorded EEG signals using modern computer facilities. The broadest and most significant domain of CEEG is represented by

methods of frequency (spectral) analysis [13]. An EEG signal is analyzed to determine its frequency characteristics, such as amplitude spectrum, cross-spectrum, coherence, etc. With respect to each characteristic, summarized spectral indices are defined in the informative frequency bands α, β, δ, γ, and θ, which enables the identification of statistical differences between the spectra. It should be noted, that the frequency nature of the mathematical methods used in CEEG reflects the fact that the initial stage of EEG analysis was associated with studying EEG rhythmic processes. Figure 16.1 shows that EEG graphoelements differ from each other not only in the values of periods but also in the peak-to-peak amplitudes of potentials. Therefore, the information contained in the graphoelements depends on both the oscillation frequency and the peak-to-peak amplitude of the potential recorded at different times. As follows from [18], the issues relating to the oscillation amplitude measuring are outside the CEEG conceptual framework. Thus, the modern electroencephalograph 'Encephalan-131-03' solves this task by working in the interactive mode only, according to a marker indicated by the operator (Figure 16.2).

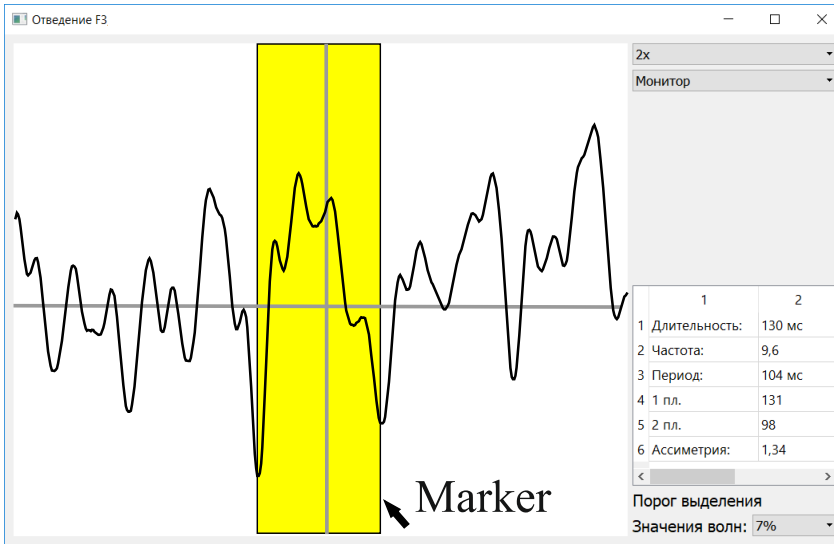


Figure 16.2. Example of measuring the peak-to-peak amplitude of the potential at a specific instant in time.

Now we briefly describe the level of the scientific scrutiny of the EEG analysis problem as of the 1970–1980s [2]. Two fundamentally different approaches can be identified here. The first one is spectral analysis. There is an assumption that the EEG is stationary or that it changes very slowly, and the spectra are computed by signal fragments of about 30 s duration. The data on different EEG patterns are lost in spectral analysis.

The second method is the analysis of individual waves, i.e. nonstationarities in the EEG. Herein, a single wave described through a set of graphoelements serves as an elementary unit. The analysis is complicated by the fact that an individual wave is examined in isolation from its context.

16.2 Structural Approach to EEG Recognition

In addition to the problem of image recognition, that has received considerable attention from researchers, studies on structural description of images have appeared in recent years [2, 4]. Image recognition suggests that each image of the object has a corresponding point in a space. In this case, objects of one class are represented by a compact set of such points. Furthermore, it is necessary to make a decision rule, according to which newly appearing points refer to one or another image.

Another challenge is to obtain a *compressed description of* one image or a whole class of images. Such tasks arise in compact encoding and transmitting, through high-usage trunks, the information about the object images that would be sufficient for reproducing the basic characteristics of the images but not their details. This task is closely related to pattern recognition since the elements of this description can be used as informative features based on which the image of the object can be reasonably assigned to one of the alphabet classes.

The structural (syntactic) approach implies that an image to be recognized (an electroencephalogram) is made up by linking together simple subimages in a particular way, similar to connecting phrases into a sentence by means of word joining. This approach can be useful only in the case when the simplest subimages selected are much easier to recognize that the entire image, i.e. the EEG. In the structural

approach to image recognition, these subimages are referred to as *primitive elements*. The rules for primitive element composition are usually defined by the so-called grammar of the image description language. It is exactly the sort of parsing that the clinician performs when visually inspecting an EEG in the process of monitoring the dynamics of potentials. The brain's bioelectrical activity is the generation of oscillations in the frequency domain of $0.3 \div 50$ Hz, which is divided into frequency bands of particular activity types: δ, θ, α, β, and γ. In the case of visual inspection, each frequency domain component is estimated by determining the frequency and the oscillation amplitude. Therefore, it is expedient to select an individual wave $\mathbf{W}_r$, $r = 0, 1, 2, \ldots$, as a primitive element because it is simple enough in its shape and observable in any EEG. As already noted, an EEG wave is defined as an oscillation fragment limited by *two* sequential local minima [2,17].

The shape of the wave, its length and peak-to-peak amplitude, as well as its temporal position within the EEG limits, are informative features of a primitive element. In this approach, an arbitrary EEG represents an ordered sequence of numbered wave-shaped primitive elements. If the current nth wave is denoted by $\mathbf{W}_r$, $r = 0, 1, \ldots, k-1$, then the entire EEG $\mathbf{G}$ can be represented analytically as a concatenation of its numbered waves:

$$\mathbf{G} = \mathbf{W}_0 \oplus \mathbf{W}_1 \oplus, \ldots \oplus \mathbf{W}_{R-1}, \tag{16.1}$$

where R is the number of waves in the EEG and $\oplus$ denotes the operation of wave concatenation. In line with the structural approach, each EEG $\mathbf{G}$ is represented as a set of its primitive elements (segments) $\{\mathbf{W}_r\}_0^{R-1}$, sequentially adjacent to each other. Following this interpretation, a system for automatic EEG classification comprises a preprocessor, a primary processing unit (PPU), and a secondary processing unit (SPU) (Figure 16.3).

This allows taking into consideration the context of successive primitive elements (segments).

The preprocessor of the system consists of a mathematical model generator $\mathbf{\Gamma} = \{\gamma(n)\}_0^{k-1}$ and a device for the segmentation of EEG $\mathbf{G}$. The signals $\{u(n)\}_0^k$ from the encephalograph output are EEG counts acquired through sampling at a time interval of t_∂ s. In order to obtain EEG oscillations in a continuous form, the samples u_n, $n = 0, 1, \ldots, k-1$, are interpolated. Figure 16.4 shows the result $\mathbf{\Gamma} = \{\gamma(n)\}_0^{k-1}$ of linear interpolation of these samples.

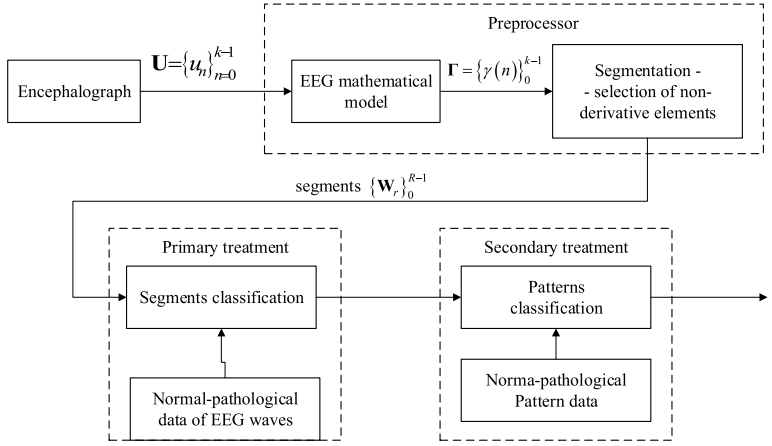


Figure 16.3. A system for automatic EEG classification based on the structural method and intended to support the clinician.

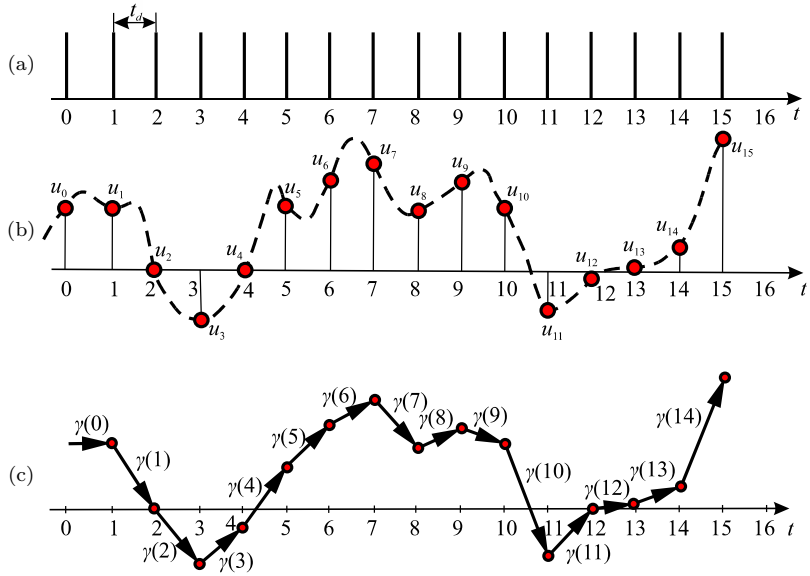


Figure 16.4. Operation of the generator of the mathematical model of the EEG being classified: (a) sampling pulses; (b) potential samples; (c) vector representation of linearly interpolated samples.

The mathematical model $\mathbf{\Gamma} = \{\gamma(n)\}_0^{k-1}$ is written as a sequence of elementary vectors (EVs)

$$\gamma(n) = u(n+1) - u(n), \quad n = 0, 1, \ldots, k-1, \tag{16.2}$$

formed up the last two potential samples. The mathematical model (16.2) describes the *contour of an* EEG pattern and allows not only obtaining the oscillation shape data but also through segmenting the contour sections between adjacent minima, the isolation of waves as EEG primitive elements $\mathbf{W}_r$, $r = 0, 1, \ldots, R-1$.

The segmentation process is performed by the second unit of the preprocessor (Figure 16.3). One of the segmentation approaches was considered in [7–9]. The primary processing unit PPU performs the function of segment automatic classification (recognition). Its input receives the vector of segments $\mathbf{W} = \{\mathbf{W}_r\}_{r=0}^{R-1}$, with $\mathbf{W}_r = \{\gamma_r(n)\}_0^{\kappa_r - 1}$. Here, R is the number of segments, and κ_r is the current segment dimensionality equal to the number of its constituent EVs $\gamma_r(n)$. Based on the developed algorithms, the informative energy, frequency, and time features are computed for each segment $\mathbf{W}_r$. The features of this dataset are then compared with their norm-pathology values in regard to the segments, and, based on the results of the comparison, each segment is subsumed under one of three classes: normal, borderline, or pathological.

The secondary processing unit, SPU, generates the results of EEG analysis. Its input receives a sequence of "letters" in the form of segments $\mathbf{W}_r$, $r = 0, 1, \ldots, R-1$, recognized by PPU. During operation, SPU singles out 'words', complexes (phenomena and patterns) from this sequence. A complex is understood as a chain of two or more waves (segments) and an informative feature of the processes occurring in the human central nervous system [16, 18]. For instance, the 'Spindle' complex formed as a chain of oscillations close to sinusoidal ones in shape and having a frequency of approximately 10 Hz is typical of a healthy individual's EEG during the awake state (segments 11–19 in Figure 16.5). Figure 16.5 also illustrates a chain of the segments $\mathbf{W}_0$, $\mathbf{W}_1$, and $\mathbf{W}_2$ — "a slow wave, a spike, a peaked wave" — which is a pathological complex. The complex "multiple spikes, a slow wave" composed of the segments $\mathbf{W}_{33} \div \mathbf{W}_{36}$ is shown in Figure 16.5. The results of such "grammatical analysis of the sentence" presented as EEG oscillations serve as the basis for a clinical report which is automatically produced by the

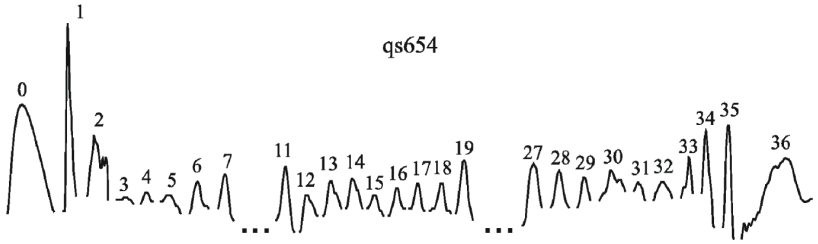


Figure 16.5. Segmentation of the EEG presented in Figure 16.1.

system (Figure 16.3). Further in this chapter, we review analytical approaches to the implementation of the system of automatic EEG classification utilizing structural methods.

16.3 Mathematical Model of the EEG

In order to solve the problem of the automation of EEG processing, the currently applied *spectral* approach underlying computerized EEG methods [13] and ensuring the frequency resolution of the signal is to be supplemented with *temporal* resolution methods whose application does not require human participation. The spectral approach employs a mathematical model written as a pulse amplitude-modulated sequence of samples $u(n)$, $n = 0, 1, \ldots, k$, acquired at a time interval t_d from the electroencephalograph output (Figure 16.4(b)). The real vector

$$\mathbf{U} = \{u(n)\}_0^{k-1} \tag{16.3}$$

is called the *impulse* model of EEG. By means of linear interpolation of the values of samples $u(n)$, we obtain the polygonal sequence

$$\mathbf{\Gamma} = \{\gamma(n)\}_0^{k-1}, \tag{16.4}$$

whose components $\gamma(n)$, indicated above as elementary vectors, are represented by complex numbers (Figure 16.6):

$$\gamma(n) = t_\partial + \mathbf{i}\Delta u(n) = |\gamma(n)| \exp\{i\psi(n)\}. \tag{16.5}$$

Here, t_d is the EEG sampling interval, $\Delta u(n) = u(n+1) - u(n)$ is the first-order difference of the digital samples from the output of the electroencephalograph, and $|\gamma(n)|$ and $\psi(n)$ are the modulus and the

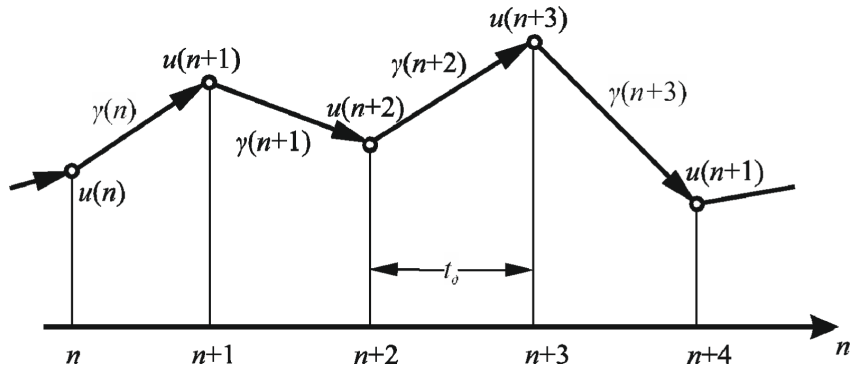


Figure 16.6. Formation of elementary vectors $\gamma(n)$ of the mathematical model of a contour.

argument of the number $\gamma(n)$, respectively. The contour of the EEG, presented as a sequence of EVs $\gamma(n)$, will be hereinafter referred to as the EEG contour model set in a unitary space C [3, 6].

16.4 Contour Model as an Element of a Unitary Space

In the following, we provide some data on the model properties, which would be useful for further analysis [9].

The points of the k-dimensional unitary space C_k coincide with the points of the space E_{2k}. The operation of the scalar product is given in C_k

$$\eta_C = (\gamma, \nu) = \gamma\nu^* = \|\gamma\|\|\nu\| \cos\varphi - i\|\gamma\|\|\nu\| \sin\varphi, \tag{16.6}$$

where φ is the angle between the vectors set by the numbers γ and ν. Based on (16.6), the norm $\|\mathbf{\Gamma}\| = \sqrt{(\mathbf{\Gamma}, \mathbf{\Gamma})}$ of the complex signal is introduced, which specifies the length of the vector. The squared norm $E_{\mathbf{\Gamma}} = \|\mathbf{\Gamma}\|^2$ defines the energy of this signal. The scalar product of the contours $\mathbf{\Gamma} = \{\gamma(n)\}_0^{k-1}$ and $\mathbf{N} = \{\nu(n)\}_0^{k-1}$, with regard to (16.6), is expressed as follows:

$$(\mathbf{\Gamma}, \mathbf{N}) = \sum_{n=0}^{k-1} \|\gamma(n)\|\|\nu(n)\| \cos\varphi_n - i \sum_{n=0}^{k-1} \|\gamma(n)\|\|\nu(n)\| \sin\varphi_n. \tag{16.7}$$

Further on, we use the superscript "∘" to indicate normalized vectors, for example, $\mathbf{\Gamma}^{\bigcirc} = \mathbf{\Gamma}/\,\|\mathbf{\Gamma}\| = \{\gamma^{\circ}(n)\}_0^{k-1}$, where $\|\mathbf{\Gamma}^{\circ}\| = 1$. A normalized scalar product (NSP) is written as

$$\eta_C^{\circ} = \mathrm{Re}\eta_C^{\circ} + i\,\mathrm{Im}\eta_C^{\circ} = (\mathbf{\Gamma}^{\circ}, \mathbf{N}^{\circ}) = (\mathbf{\Gamma}, \mathbf{N})/\,\|\mathbf{\Gamma}\|\|\mathbf{N}\|$$
$$= \sum_{n=0}^{k-1} \cos\varphi_n - i\sum_{n=0}^{k-1} \sin\varphi_n, |\eta_C^{\circ}| \le 1. \tag{16.8}$$

Here, φ_n is the angle between EVs γ_n and ν_n, and φ is the angle between $\mathbf{\Gamma}$ and $\mathbf{N}$. As noted in [3], due to the presence of the imaginary part, the scalar product of vectors in C_k is more *informative* than the scalar product in space R. This allows unique determination of the angle between the vectors $\mathbf{\Gamma}$ and $\mathbf{N}$.

$$\varphi = \arctan(\mathrm{Im}\,\eta_C^{\circ}/\mathrm{Re}\,\eta_C^{\circ}). \tag{16.9}$$

The real part $\varepsilon = \mathrm{Re}\,\eta_C^{\circ}$ of NSP (16.8), which is equal to $\cos\varphi$, serves as a measure of similarity of the normalized vector signals $\mathbf{\Gamma}^{\circ}$ and $\mathbf{N}^{\circ}$ and reaches its maximum when $\varphi = 0$. It follows from (16.9), $\varepsilon = \varepsilon_{\max}$, with $\mathrm{Im}\,\eta_C^{\circ} = 0$. In this case,

$$\varepsilon_{\max} = |\mathrm{Re}\,\eta_C^{\circ} + i\,\mathrm{Im}\,\eta_C^{\circ}|, \tag{16.10}$$

i.e. the maximum value of the similarity measure for vectors $\mathbf{\Gamma}^{\circ}$ and $\mathbf{N}^{\circ}$ is equal to the modulus of NSP of these vectors.

Example. Figure 16.7 shows the images of two contours $\mathbf{\Gamma} = \{\gamma(0);\gamma(1)\} = \{1+1,9i; 3-3i\}$ and $\mathbf{N} = \{\nu(0);\nu(1)\} = \{1+2,5i; 2-4i\}$. There is little visible difference between the contours. NSPs of the contours are equal to $\eta_{C_1}^{\circ} = \mathrm{Re}\,\eta_C^{\circ} + i\,\mathrm{Im}\,\eta_C^{\circ} = \{0.975+0.218i\}$.

The similarity measure ε_R in a real space is equal to $\mathrm{Re}\eta_C^{\circ} = 0,957$, while in a unitary space, it notably approaches the maximum value equal to one: $\varepsilon_C = \varepsilon_{\max} = |0,957+0,218i| = 0,981$. The lower value of the measure of similarity between the vectors $\mathbf{\Gamma}$ and $\mathbf{N}$, which is equal to $\varepsilon_R = 0,957$, was due to its dependence on the angle $\varphi = \arctan 0,218/0,957 = 12,8°$, which is typical of specifying vectors in space R. With the introduction of vectors $\mathbf{\Gamma}$ and $\mathbf{N}$ into the unitary space C, invariant to vector rotation, the similarity measure value grew up to $\varepsilon_C = \varepsilon_{\max} = 0,981$.

The EEG contour model $\mathbf{\Gamma} = \{\gamma(n)\}_0^{k-1}$ introduced as a sequence of waves of this oscillation has significant advantages over the

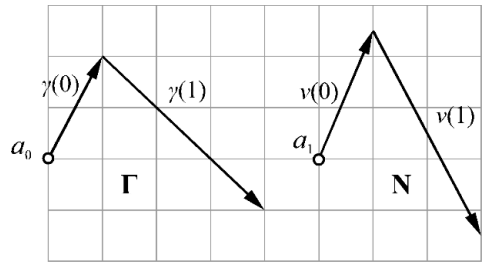


Figure 16.7. Determination of the measure of similarity between the contours (waves) $\mathbf{\Gamma}$ and $\mathbf{N}$ in unitary and real vector spaces.

practically applied discrete impulse model $\mathbf{U} = \{u(n)\}_0^{k-1}$ of the EEG. In particular, the contour model of the EEG

(1) presents the waveform analytically,
(2) ensures the isolation of primitive elements (waves, segments),
(3) enables the analysis of oscillation frequency and time properties from common analytical perspectives,
(4) generates a rotationally invariant similarity measure of two primitive elements.

16.5 Segmentation of the EEG

Segments $\mathbf{W}_r$, $r = 0, 1, \ldots, R-1$, of the EEG contour have pulse shapes determined by individual oscillation waves. Their analysis makes it possible not only to compute the averaged process characteristics previously acquired by spectral-correlation methods but also to form an EEG resolved along the time axis. Each segment (primitive element) of the resolution represents a separate current waveform whose shape is available for detailed analysis. To illustrate this, we consider an EEG fragment comprising several periods of a sinusoid (Figure 16.8).

The procedure of EEG segmentation was considered in [7, 8]. In the following, it is exemplified on a real EEG (Figure 16.9(a)). It is necessary to isolate oscillation fragments as pulses $1 \ldots 6$. The pulse shape is determined by the interference of potential oscillations in the frequency bands $\delta, \theta, \alpha, \beta$, and γ; which results in the appearance of fragments containing local extrema within the pulse envelope and,

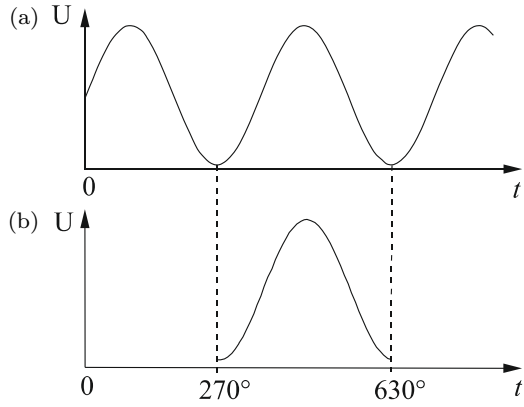


Figure 16.8. Selecting the shape of the EEG element for temporal resolution: (a) sinusoidal oscillation; (b) type of the EEG element (segment, pulse) for temporal resolution.

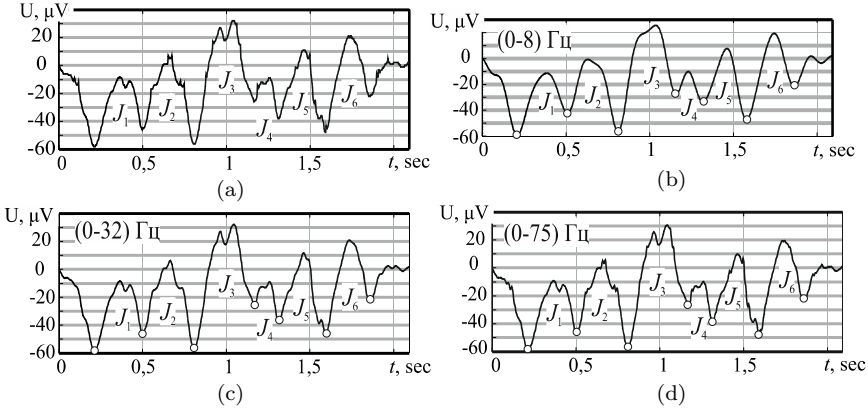


Figure 16.9. Changes in the oscillation pulse shapes when changing the bandpass Δf of the bandpass filter: (a) original oscillation; (b) $\Delta f = (0 \div 8)$ Hz; (c) $\Delta f = (0 \div 32)$ Hz; (d) $\Delta f = (0 \div 75)$ Hz.

consequently, in the ambiguity of positions of the pulse start and endpoints. To reduce the impact of this factor, the iterative principle of the segmentation algorithm was used.

Each iteration includes two operations: the low-frequency filtering of the EEG contour, and the recording of the points of consecutive global minima. Figure 16.9 illustrates the result of the first iteration.

Following filtering in the band $(0 \div 8)$ Hz, only low-frequency components of the contour remain, which allows us to determine the points of its consecutive global minima unambiguously (Figure 16.9(b)). During the next iteration, the filter band slightly increases, which causes minor distortions in the pulse shapes. Since a rough estimation of the position of pulse boundaries was performed in the previous iteration, it is now possible to refine it. Figure 16.9 shows the EEG after filtration in the band $(0 \div 30)$ Hz, and Figure 16.9(d) in the band (0–75) Hz.

Comparison of the images of the original EEG (Figure 16.9(a)) with the final one (Figure 16.9(d)) suggests that there are no visible distinctions between the images. As shown in Figure 16.9(c)), the wave boundaries, marked with circles, are in good agreement with the accepted rules. The sampled EEG consists of segments $\mathbf{W}_r$, $r = 0, 1, \ldots, R-1$, each presented in the contour form:

$$\mathbf{W}_r = \{\gamma(n)\}_{n_r}^{n_r+k_r-1}, \tag{16.11}$$

where r is the sequence number of a segment. The segment $\mathbf{W}_r$ sets the shape of the current rth EEG wave and is named an *isolated* segment. It characterizes a separate r wave without taking into account its *temporal position* within the analyzed EEG epoch. Of higher informative value is the *modified* segment $\mathbf{W}_{r,m}$, whose isolated segment$\mathbf{W}_r$ is supplemented on the right and left by the zeros of EVs located outside the segment $\mathbf{W}_r$ (Figure 16.10(b)). As a result, it becomes possible to evaluate the CNS activity not only within the time interval of the segment $\mathbf{W}_{r,m}$ but also to assess the *effect of* any wave on the entire EEG image.

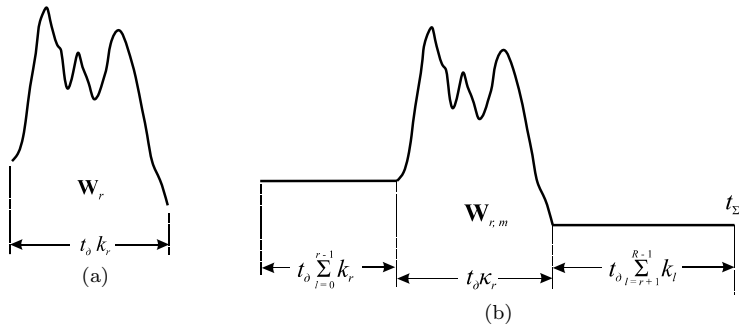


Figure 16.10. Types of segments: (a) isolated segment; (b) modified segment, t_Σ— epoch length of the EEG being analyzed.

16.6 Spectral Analysis of the EEG

16.6.1 *Orthogonal basis in a unitary space*

The condition for the orthogonality of signals $\mathbf{\Gamma}$ and $\mathbf{N}$ in a unitary space is as follows:

$$\mathrm{Re}(\mathbf{\Gamma}, \mathbf{N})_{C_k} = 0; \quad \mathrm{Im}(\mathbf{\Gamma}, \mathbf{N})_{C_k} = 0.$$

The first expression is the condition for the orthogonality of vector signals in space R_{2k}, and the second one sets an additional condition for specifying signals in the unitary space C_k. The complex-valued signals $\mathbf{\Gamma}_m$, called *elementary contours* (ECs), meet this condition [6]:

$$\mathbf{\Gamma}_m = \{\gamma_n(m)\}_{n=0}^{k-1} = \exp\left\{i\frac{2\pi}{k}mn\right\}_{n=0}^{k-1}, \tag{16.12}$$

where $m = 0, 1, \ldots, k-1$, the EC order. An EC is a $k-$sided regular polygon, whose sides are normalized EVs $\gamma(n)$, $n = 0, 1, \ldots, k-1$ (Figure 16.11).

16.6.2 *DFT of the EEG contour*

Discrete Fourier transformation $\mathbf{P}_{\mathbf{\Gamma}} = \{\rho_{\mathbf{\Gamma}}(m)\}_0^{k-1}$ of a vector signal $\mathbf{\Gamma} = \{\gamma(n)\}_0^{k-1}$, formed on the basis of a periodic signal $\mathbf{U} = \{u(n)\}_0^{k-1}$ with the period k is written as follows

$$\begin{aligned}\rho_{\mathbf{\Gamma}}(m) = (\mathbf{\Gamma}, \mathbf{\Gamma}_m) &= \sum_{n=0}^{k-1}\gamma(n)\exp\left\{-i\frac{2\pi}{k}mn\right\} \\ &= \sum_{n=0}^{k-1}(t_d + i\Delta u(n))\exp\left\{-i\frac{2\pi}{k}mn\right\} \\ &= \rho_{t_d}(m) + i\rho_{\Delta u}(m), \quad m = 0, 1, \ldots, k-1.\end{aligned} \tag{16.13}$$

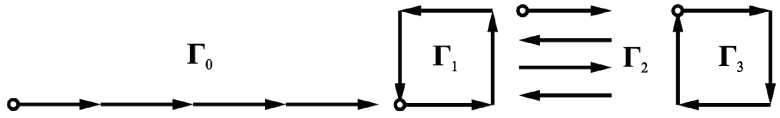


Figure 16.11. Family of mutually orthogonal elementary contours $\{\mathbf{\Gamma}_m\}_0^3$ with dimensionality $k = 4$.

Here,

$$\rho_{t_d} = \sum_{m=0}^{k-1} t_d \exp\left\{-i\frac{2\pi}{k}mn\right\} = \begin{cases} kt_d & for \quad m=0; \\ 0 & for \quad other\ k \end{cases},$$
$$\rho_{\Delta u}(m) = \sum_{n=0}^{k-1}(u(n+1)-u(n))\exp\left\{-i\frac{2\pi}{k}mn\right\}. \tag{16.14}$$

In the latter expression, the summation symbol precedes the difference of the shifted by one segment sequence of signals from the electroencephalograph output. Based on the properties of the spectra of such sequences [6], we obtain

$$\rho_{\Delta u}(m) = \rho_{\mathbf{U}}(\gamma_1(m) - \gamma_0(m)), \quad m = 0, 1, \ldots, k-1, \tag{16.15}$$

where $\gamma_1(m)$ and $\gamma_0(m)$ are EVs of the ECs $\mathbf{\Gamma}_1$ and $\mathbf{\Gamma}_0$:

$$\mathbf{\Gamma}_1 = \{\gamma_1(m)\}_0^{k-1} = \left\{\exp\left\{i\frac{2\pi}{k}\cdot 1\cdot n\right\}\right\}_0^{k-1};$$
$$\mathbf{\Gamma}_0 = \{\gamma_1(0)\}_0^{k-1} = \left\{\exp\left\{i\frac{2\pi}{k}\cdot 0\cdot m\right\}\right\}_0^{k-1}. \tag{16.16}$$

Let us consider the issue of the correlation between the spectrum of the EEG epoch contour $\mathbf{\Gamma}$ and the spectra of isolated $\{\mathbf{W}_r\}_0^{R-1}$ and modified $\{\mathbf{W}_{r,m}\}_0^{R-1}$ segments. As follows from expression (1), the epoch $\mathbf{G}$ is expressed through the concatenation of segments $\mathbf{W}_r$, $r = 0, 1, \ldots, R-1$. As segment dimensionalities *do not coincide* with the dimensionality of the contour, then the spectrum of the sum of isolated segments is not equal to the sum of spectra of these segments, i.e.

$$\mathbf{P}_{\mathbf{\Gamma}} \neq \sum_{r=0}^{R-1} \mathbf{P}_{\mathbf{W}_r}.$$

It is significantly easier to express the spectrum $\mathbf{P}_{\mathbf{\Gamma}}$ of the contour $\mathbf{\Gamma}$ through the spectra of modified segments whose dimensions are equal to the same value k:

$$\mathbf{P}_{\mathbf{\Gamma}} = \sum_{r=0}^{R-1} \mathbf{P}_{\mathbf{W}_{r,m}}. \tag{16.17}$$

16.6.3 *The correlation between the spectrum of the contour* $\mathbf{\Gamma}$ *and the spectrum of output samples of the electroencephalograph*

Let us determine the relationship between the spectrum of real samples $\mathbf{U} = \{u(n)\}_0^{k-1}$, representing the EEG as an element of linear space E, and the spectrum of the sequence of complex-valued vectors $\mathbf{\Gamma} = \{\gamma(n)\}$, which represent the EEG as an element of the unitary space $\mathbf{C}$. The transformation will allow obtaining the spectrum $\mathbf{P_\Gamma}$ of the contour $\mathbf{\Gamma} = \{\gamma(n)\}_0^{k-1}$ through the spectrum $\mathbf{P_U}$ of the output samples of the encephalograph. Taking into account expressions (16.15) and (16.16), we get

$$\mathbf{P_\Gamma} = \{\rho_\mathbf{\Gamma}(m)\}_0^{k-1} = \begin{cases} kt_d & npu\ m = 0; \\ i\rho_\mathbf{U}(m)[\gamma_1(m) & npu\ m = 1, 2, \ldots, k-1. \\ \quad -\gamma_0(m)] & \end{cases} \tag{16.18}$$

Example. The samples of the encephalograph output signal of a real EEG fragment, with a sampling interval $t_d = 0,002\,\text{s}$, are equal to

$$\mathbf{U} = \{0; 1,309; 2,154; 2,488; 2,295; 1,595; 0,437; -1,099\}.$$

In accordance with the expression (16.5), the contour model takes the following form:

$$\mathbf{\Gamma} = \{0,002 + 1,309i; 0,002 + 0,844i; 0,02 + 0,334i; 0,002 - 0,192i; \\ 0,002 - 0,701i; 0,002 - 1,158i; 0,002 - 1,536i; 0,002 + 1,099i\}.$$

The spectrum $\mathbf{P_\Gamma}$ of the contour is equal to

$$\mathbf{P_\Gamma} = \{0,016; 2,372 + 4,339i; -1,221 + 1,811i; -1,367 - 0,319i; \\ -1,186i; 1,367 - 0,319i; 1,221 + 1,811i; -2,372 + 4,339i\}.$$

Similarly, we calculate the spectrum of the EEG impulse model:

$$\mathbf{P_U} = \{9,179; -5,033 - 4,052i; -0,295 - 1,516i; 0,443 - 0,617i; \\ 0,593; 0,443 + 0,617i; -0,295 + 1,516i; -5,033 + 4,052i\}. \tag{16.19}$$

Based on the expression (16.18), the spectrum $\mathbf{P_U}$ of the EEG fragment $\mathbf{U}$, deduced from the spectrum $\mathbf{P_\Gamma}$ of its contour model $\mathbf{\Gamma}$, is written as

$$\mathbf{P_U} = \{9,179; -5,033 - 4,052i; -0,295 - 1,516i; 0,443 - 0,617i; \\ 0,593; 0,443 + 0,617i; -0,295 + 1,516i; -5,033 + 4,052i\}. \tag{16.20}$$

The expressions obtained (16.19) and (16.20) show that the spectrum $\mathbf{P_U}$ of the original real sequence $\mathbf{U}$, with the exception of the zero sample, coincides with the expression for the spectrum of EEG set by its contour model. This result confirms the *complete adequacy of* the EEG contour model introduced in this study and its original impulse model.

16.7 Problems of Computer Analysis of the EEG

The task of EEG computer processing consists in providing the medical professional with EEG information in a readily understandable format including the detection of deviations from the norm [13]. Computerized EEG (CEEG) exploits the pulse model $\mathbf{U} = \{u(n)\}_0^{k-1}$ of discrete samples from the encephalograph output as the mathematical model of the oscillation. Methods based on fast Fourier transformation (FFT) underlie approaches to the analysis of oscillations [13]. The state of the art of EEG research is characterized by a lack of information on a universally accepted mechanism for generating potential oscillations recorded as EEG, as well as on the laws of probability distribution relating to the noises and interferences that accompany these oscillations. These factors hamper the synthesis, in accordance with the theory of signal processing, of quasi-optimal algorithms for EEG signal processing. Other problems associated with this task should be noted as well:

1. EEG is a globally non-stationary signal that reflects the activity of the central nervous system under the conditions of a continuous stream of incoming data about the state of internal organs and the environment. In this context, the application of spectral-correlation approaches is associated with conceptual constraints that lead to a decrease in the reliability of the results obtained [13].

2. The spectral-correlation methods used in CEEG are of an integral nature due to the fact that the results are generated by averaging with certain types of weights of *all* samples of the EEG under analysis. In line with this approach, the results obtained would be satisfactory only in the case of individual oscillation fragments containing information of the same nature. In fact, the same EEG segments carry heterogeneous brain activity information at different time instances, and the information is lost under averaging.
3. Despite the obvious advantages of CEEG, the *problem of the automatic formation of a* clinical conclusion cannot be solved from the perspective of this approach, because the *detection of* each complex in the EEG under analysis is to be preliminarily performed by a clinician. For example, as Figure 16.2 shows, even measuring such a commonly used EEG parameter as the peak-to-peak amplitude at a required time lies outside the frequency analysis paradigm [18].
4. Support provided to a clinician for performing the visual inspection of EEG. Due to the human ability to solve hard-to-formalize tasks, such as pattern recognition or scene analysis, visual analysis has a lot of advantages over EEG [18].

Listed above are several problems of the EEG analysis which reduce the validity and reliability of the results obtained. Existing mathematical apparatus of the CEEG implements the spectral-correlation approach to signal analysis and has fundamental limitations in its application when the signal is non-stationary, and the nature of the information it contains varies substantially within an EEG epoch. Previously, a structural approach to signal analysis was proposed, which is used, for instance, for processing acoustic signals, and is less sensitive to the peculiarities of such a signal as EEG. The approach allows for not only frequency resolution but also temporal resolution of the signals under analysis. This ensures the availability of data on EEG individual fragments, which is under averaging when only frequency methods are applied; besides, the negative effect associated with the global nonstationarity of the EEG as a signal declines significantly. The use of the structural approach to EEG analysis required replacing the impulse model $\mathbf{U} = \{u(n)\}_0^{k-1}$ with a model of the contour type $\mathbf{\Gamma} = \{\gamma(n)\}_0^{k-1}$, as well as the substantiation of the EEG waves $\{\mathbf{W}_r\}_0^{R-1}$ as primitive elements,

i.e. as the simplest fragments of an oscillation which retain its properties; EEG decomposition into segments, i.e. individual primitive elements, was also entailed. These issues were discussed above when substantiating the adequacy of the mathematical models $\mathbf{U}$ and $\mathbf{\Gamma}$. The results obtained allow proceeding to the issues of EEG automatic processing. By the latter we mean a situation in which the doctor is provided with calculated ready-made solutions, based on which, as well as his/her experience, intuition, and aptitude for non-formalized decision-making, the specialist will prepare a medical report. As follows from the system for EEG automatic classification for clinician support (Figure 16.3), these solutions are generated by the subsystems conventionally termed as *units for primary and secondary processing* of the segmented oscillation (PPU and SPU).

The system for automatic classification of the epoch $\mathbf{G}$ takes a sound decision on subsuming an EEG to one of three classes: normal, borderline, or pathological [17, 19] (Figure 16.12). To this end, the decision-maker of the system compares the *computed* parameters of the oscillation with the parameter values permissible for each class.

The system is implemented in accordance with the following principles:

Principle 1. Absence of manual operations in the analysis of the EEG.
Principle 2. Structural approach to the EEG classification procedure.
Principle 3. The EEG is set by the contour mathematical model.
Principle 4. Generation of the informative features for classifying the EEG is based on the parameters of its segments.
Principle 5. Stagewise nature of the EEG classification procedure.

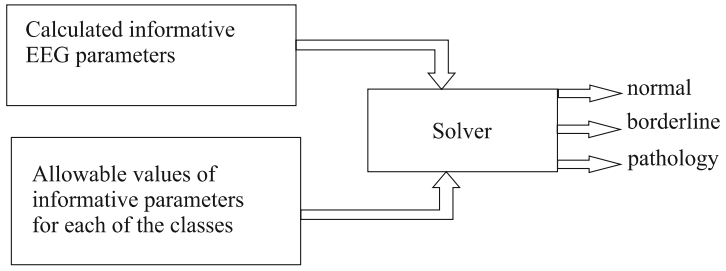


Figure 16.12. Generalized structure of the EEG classifier.

Stage I: *Preprocessing of the EEG:*

1.1 Selection of epoch $\mathbf{G}$; formation of contour $\mathbf{\Gamma} = \{\gamma(n)\}_0^{k,-1}$;
1.2 Segmentation of the epoch $\mathbf{G}$: $\mathbf{W} = \{W_r\}_{r=0}^{R-1}$

Stage II: *Primary processing of the epoch segments.*

2.1 Evaluation of the frequency and time parameters of the segment $\mathbf{W}_r$ $r = 0, 1, \ldots, R-1$
2.2 Producing an alphabet of values of the informative parameters.
2.3 Classification of segments.

Stage III: *Secondary processing of the segments of the epoch*

3.1 Detection of patterns (sets of segments).
3.2 Analysis of the patterns.
3.3 Creating a diagnostic database.

To exemplify the automatic classification of real EEGs, the results of classifying the EEG 'Autumn 2016' of a healthy patient of the Republican Clinical Hospital of the Republic of Mari El (October, 2016) are given in the following. The epoch $\mathbf{G}$ of this EEG is shown in Figure 16.13(a). The epoch contains 1064 samples from the output of

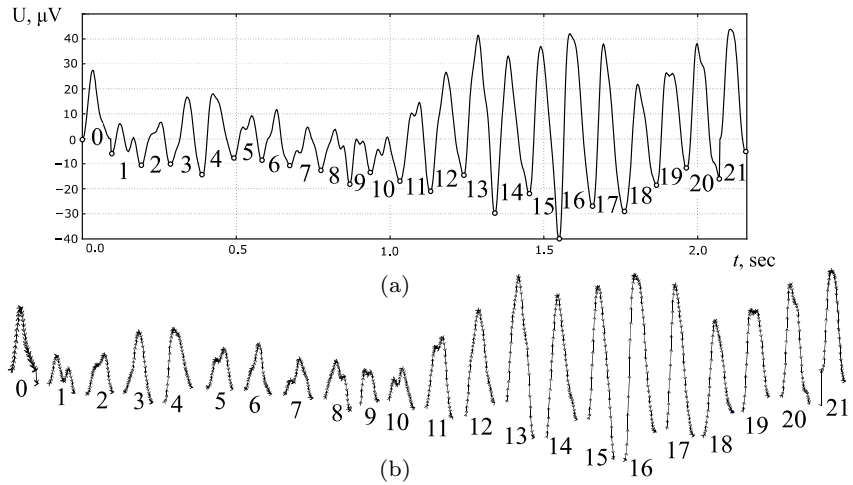


Figure 16.13. Classified fragment of the EEG of a healthy patient: (a) type of fragment; (b) segments.

the encephalograph; the sampling frequency is $\mathbf{F}_d = 500\Gamma u$ ($t_s = 2 \cdot 10^{-3}$ sec). The EEG contour is indicated as $\mathbf{\Gamma} = \{\gamma(n)\}_0^{1062}$. Following the decomposition, the EEG was presented as isolated segments $\mathbf{W} = \{W_r\}_{r=0}^{21}$ (Figure 16.13(b)), $\mathbf{W}_r = \{\gamma(n)\}_{n_r}^{-n_r+k_r-1}$, where n_r and n_{r+k_r-1} are the numbers of the starting and the ending EVs of the segment, respectively, $r = 0, 1, \ldots, 21$.

16.8 Primary Processing of Segments in an Epoch

A vector signal in the form of segments $\mathbf{W} = \{\mathbf{W}_r\}_{r=0}^{21}$, where $\mathbf{W}_r = \{\gamma_r(n)\}_0^{k_r-1}$ comes to the input of the primary processing unit (PPU). Here, k_r is the dimensionality, and $\gamma(n)$ is the EV of the contour of the segment envelope. The task for the PPU is to classify each of the segments on the basis of computing its energy parameters.

16.8.1 *Classification of segments on the basis of belonging to a particular frequency band*

The following upper and lower limits have been determined in regard to the informative frequency bands:

$$\begin{gathered}\delta \to (0.35 - 4.2)\,\text{Hz}; \quad \theta \to (4.6 - 8)\,\text{Hz}; \quad \alpha \to (8 - 14)\,\text{Hz};\\ \beta_1 \to (14 - 25)\ \text{Hz}; \quad \beta_2 \to (25 - 40)\ \text{Hz}; \quad \gamma \to (40 - 70)\,\text{Hz}.\end{gathered} \tag{16.21}$$

The rule for subsuming the segments of EEG 'Autumn 2016' being analyzed to one of the frequency bands is based on calculating the fractions of energy of the modified segment $\mathbf{W}_{r,m}$ with regard to each of the bands (16.21).

$\hat{E}_z$ indicates *the energy fraction* of segment $\mathbf{W}_r$ in the *frequency* band $z = \delta \wedge \theta \wedge \alpha \wedge \beta_1 \wedge \beta_2 \wedge \gamma$, and $\hat{E}_{r,\max}$ indicates the value of the maximum energy fraction for the segment $\mathbf{W}_r$ with the maximum energy fraction in the frequency ranges (16.21). The current segment $\mathbf{W}_r$ is considered to belong to the frequency band z, if its energy fraction in this range is maximum. It is often the case that a current segment has the energy fraction values of the current segment approach maxima in several frequency bands simultaneously. In such a situation, based on the frequency features of the segment, we will subsume

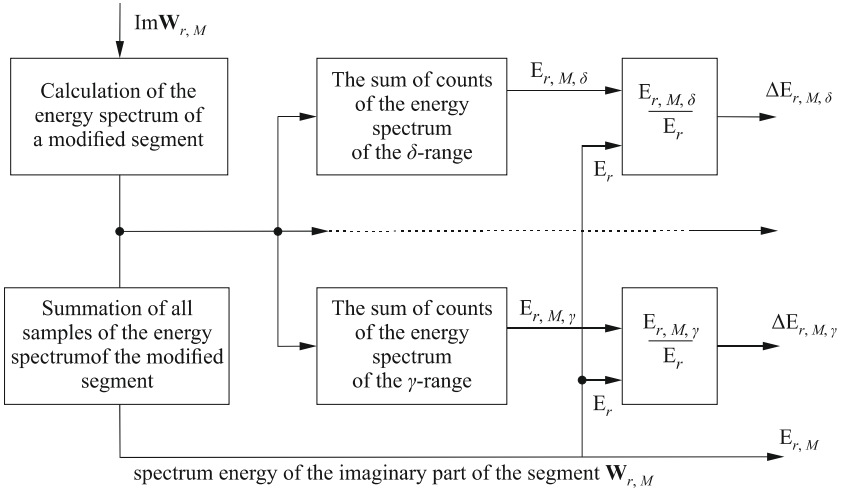


Figure 16.14. Calculating the segment $\mathbf{W}_{r,M}$ energy based on the structural approach.

it to several bands, if its fractional energy in these bands is no more than 30% less than the maximum fraction. Figure 16.14 shows the structure of the EEG energy calculation, and Table 16.1 contains the calculation results. The main table fields are represented by columns 1–6 and contain the energy values for the segments $\mathbf{W}_{0,m} \div \mathbf{W}_{21,m}$ in the frequency bands (16.21).

The sequence numbers of the segments are given in column 0. For example, line 8 of Table 16.1 corresponds to the segment $\mathbf{W}_8$. Accordingly, the segment energy fractions in these bands are equal to 0.134, 0.042, 0.263, 0.231, 0.169, and 0.112. Column 9 of the table contains, in regard to each of the segments, the names of frequency bands with the maximum fractional energy. As the table shows, the greatest energy is concentrated in the α band. The convention is that the EEG of a segment is active in the frequency band z provided that

$$\hat{E}_{\mathbf{W}_{r,m/z}} \geq E_{\Pi}. \tag{16.22}$$

With the threshold level $E_{\Pi} = 0.1$ (10% of the total energy of the segment), all segments of the epoch $\mathbf{G}$ are active in the θ and α bands only. Over 65% of the total energy of the epoch is concentrated in these bands, and the α band energy is 1.35 times greater than the energy in the θ band. The classification results for the segments $\mathbf{W}_0 \div \mathbf{W}_2$ are presented in column 8 of Table 16.1.

Table 16.1. Distribution of the energy of modified segments across the frequency bands of epoch G.

	The main frequency bands of the segment							
r	Delta band	Theta band	Alpha band	Beta1 band	Beta2 band	Gamma 1 band	Sum	Classification results
0	1	2	3	4	5	6	7	8
0	0.327	0.364	0.222	0.033	0.006	0.0001	1	δ, θ
1	0.068	0.102	0.193	0.373	0.181	0.055	1	β *1*
2	0.009	0.087	0.42	0.26	0.139	0.055	1	α
3	0.02	0.183	0.525	0.208	0.022	0.029	1	α
4	0.12	0.285	0.384	0.138	0.045	0.019	1	α
5	0.043	0.179	0.38	0.203	0.137	0.038	1	α
6	0.005	0.095	0.453	0.338	0.053	0.037	1	α
7	0.148	0.036	0.367	0.247	0.103	0.068	1	α
8	0.135	0.043	0.263	0.231	0.168	0.112	1	α
9	0.178	0.108	0.029	0.312	0.178	0.133	1	β
10	0,304	0,01	0,154	0,225	0,112	0,067	1	α, δ
11	0.004	0.129	0.468	0.238	0.093	0.046	1	α
12	0.092	0.293	0.454	0.092	0.028	0.029	1	α
13	0.178	0.346	0.344	0.065	0.04	0.019	1	θ, α
14	0.049	0.278	0.471	0.126	0.025	0.034	1	α
15	0.066	0.258	0.434	0.135	0.057	0.033	1	α
16	0.11	0.313	0.389	0.105	0.043	0.029	1	θ, α
17	0.039	0.242	0.485	0.159	0.037	0.025	1	α
18	0.002	0.133	0.541	0.229	0.035	0.04	1	α
19	0.208	0.329	0.309	0.084	0.049	0.014	1	θ, α
20	0.1826	0.369	0.349	0.063	0.024	0.008	1	θ, α
21	0.3108	0.374	0.255	0.052	0.007	0.0007	1	θ, δ

16.8.2 *Key results of the frequency analysis*

1. The EEG 'Autumn 2016', similar to other oscillations of potentials on the human scalp, is distinguished by the significant non-stationarity induced by the brain activity dynamics. Due to this, the CEEG method used to analyze the brain state produces smoothed, averaged quantitative results. The quantitative data obtained by the structural approach characterize the potential values at specific time points corresponding to particular segments; therefore, the nonstationarity of the EEG oscillation is *not a significant interfering factor* in the analysis of the signal

frequency properties. As follows from the analysis of the energy of the modified segments $\mathbf{W}_0 \div \mathbf{W}_{21}$, according to this parameter, the EEG being analyzed is divided into regions J_1 and J_2. Region J_1 consists of the segments $\mathbf{W}_1 \div \mathbf{W}_{10}$ with an average energy of 2600 μV^2, and region J_2 comprises segments $\mathbf{W}_{11} \div \mathbf{W}_{21}$ with an average energy of 22000 μV^2.

2. The substantiated classification of each of the segments $\{\mathbf{W}_r\}_0^{21}$ was carried out according to the segment belonging to the informative frequency bands. Based on classification results, 18 out of 22 segments were subsumed to the α band. Notably, some of the segments exhibited high energy levels in not one but two frequency bands. The second band that most often exhibited these features was the θ band.
3. In order to analyze EEG frequency characteristics using both CEEG and structural analysis, the same statistics were selected in the form of spectrum $\mathbf{P_U}$ of the counts from the output of electroencephalograph (EE). The spectral energy values of all segments $\{\mathbf{W}_r\}_0^{21}$ exactly coincided with the total fractional energy of the EEG in all frequency bands, which confirms the validity of results obtained by the structural method.

16.8.3 *Classification of the EEG 'Autumn 2016' segments by their time parameters*

To make a decision on the class of an isolated segment $\mathbf{W}_r$, $r = 0, 1, \ldots, 21$, the following informative features of the segment pulse are taken into account: the peak-to-peak amplitude, the base-level length and the length at 0.707 relative to the amplitude peak, the shape of the peak, the measure of similarity with a sinusoidal shape waveform, the degree of symmetry, the angle of inclination, as well as the index, the pulse shape status, and the pulse peak-to-peak amplitude values in the α and β bands. To classify an isolated segment $\mathbf{W}_r$, the quantitative parameter values are compared with the data on the normative parameters for the normal, borderline, and pathological EEG, as specified in the monographs [17, 19]. The peak-to-peak amplitude of each segment in the EEG under analysis is no less than 4 μV, which is several times greater than the encephalograph self-noise level [18]; therefore, the segments will be classified.

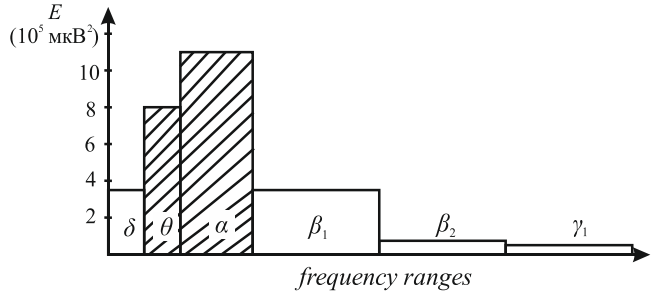


Figure 16.15. Diagram of the EEG segment energy distribution across the frequency bands (computerized EEG).

16.8.4 *Temporal distribution of the EEG energy and amplitudes*

In compliance with [17], an EEG of an adult is considered "normal" if it is mainly represented *by* α and β rhythms. The δ and θ band waves of the "normal" EEG are not expected to significantly exceed the dominant rhythm. The diagram produced by the CEEG method (Figure 16.15) presents the energy distribution across the frequency bands; apparently, the distribution conforms to a normal EEG. 55% of the total EEG energy equal to $0.28\,\mathrm{mV}^2$ is concentrated in the α and β bands, with the maximum energy value of $0.11\,\mathrm{mV}^2$ allocated in the α band. In the case of the *structural* approach, each of the 22 isolated segments is subject to analysis, i.e. EEG waves are measured and analyzed in sequence. The result averaged for all segments *coincides* with the results obtained by CEEG, which allows us to classify this oscillation as "normal". However, this conclusion is not supported by the differential analysis of the segments which was carried out in line with the structural approach. The diagram (Figure 16.16) shows the positions of all EEG segments, their energy values, as well as the frequency bands they belong to.

As the diagram shows, the EEG energy level varies stochastically within the wide time and frequency ranges. As in the CEEG, the α band contains the maximum amount of the segment energy; nevertheless, not all of the segments can be categorized as belonging to the alpha band: the maximum energy is observed in the θ band and, for one of the segments, in the α and β_1 bands. Approximately 20% of the EEG energy is allocated beyond the α band.

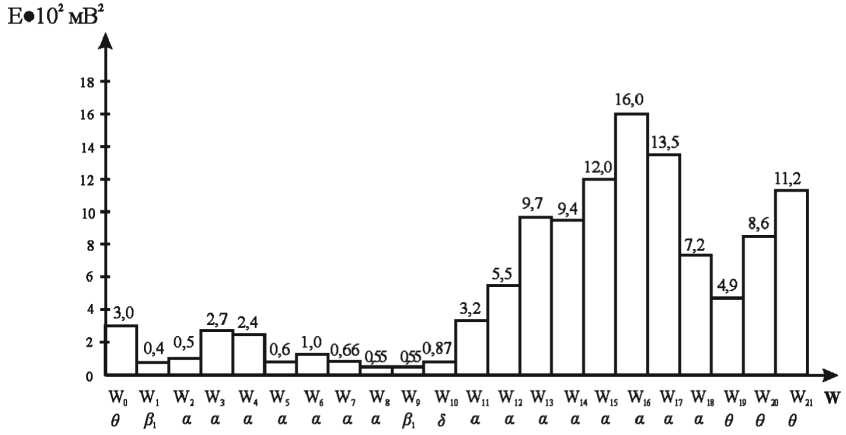


Figure 16.16. Diagram of the EEG segment energy distribution across frequency bands (the structural approach).

Thus, the *average values* of the parameters acquired using the proposed method coincide with the averages obtained by CEEG. The average values of the parameters allow for the conclusion that the EEG 'Autumn 2016' is normal. However, the results of the analysis of the energy distributions and potential levels obtained on the basis of the structural approach suggest that, according to the behavior of these parameters, the EEG 'Autumn 2016' belongs to the *borderline class*.

16.8.5 *Classification of segments by the vertex angle measure*

Scientific publications available do not specify any standards or criteria for measuring the degree of sharpness of the segment pulse top; the judgment "the waveform top is sharp" is made subjectively, based solely on the visual inspection of the EEG. Quantitatively, the boundary values of angles φ at the vertex were determined by expertise: a segment with the vertex angle φ is considered as 'normal' at $\varphi \geq 25°$, as 'boundary' at $10° \leq \varphi < 25°$, and as 'pathological' at $\varphi < 10°$. The analysis of values of the angle φ for the EEG 'Autumn 2016' indicated that 10 out of 22 EEG segments exhibited the boundary values, and thus, based on this parameter, the EEG 'Autumn 2016' belongs to the '*boundary' class*.

16.8.6 *Characteristic of the top shape and length of segment pulses*

The length values $\tau_{0.707}$ and τ_{ax} at the level of 0.707 relative to the amplitude peak and to the base depend on the segment energy distribution across the informative frequency bands. As the diagram in Figure 16.15 shows, approximately 70% of the oscillation total energy is concentrated in the α and θ bands, predominantly in the α band. These are lower frequency bands compared with β_1, β_2, and γ *bands*. Given the large number of segments belonging to the α band, the τ_{est} values are expected to be concentrated about its mean value. This assumption has been substantiated by the calculations: the mean square deviation (MSD) of the length τ_{est} for the EEG 'Autumn 2016' pulses is equal to 0.012 with a mean of 0.09. An abnormal situation is observed in regard to segment $\mathbf{W}_0$, too, whose energy in the β range exceeds its combined energy in the α and θ bands approximately by a factor of 4. The change in the shape of the pulse's upper part, which involves the appearance of several peaks, results in the helmet-like shape of the pulse. Such a segment is formed by the EEG "slow" wave, whose length is determined by the value τ_{ax} (Figure 16.17). Individual narrow impulses on the top of the segment expand significantly the frequency range of the "slow" wave. The helmet-like appearance of the tops of segments $\mathbf{W}_1$ and $\mathbf{W}_{10}$ resulted in the fact that 72% of the energy of segment $\mathbf{W}_1$ is observed not in the α band but in the β_1 band.

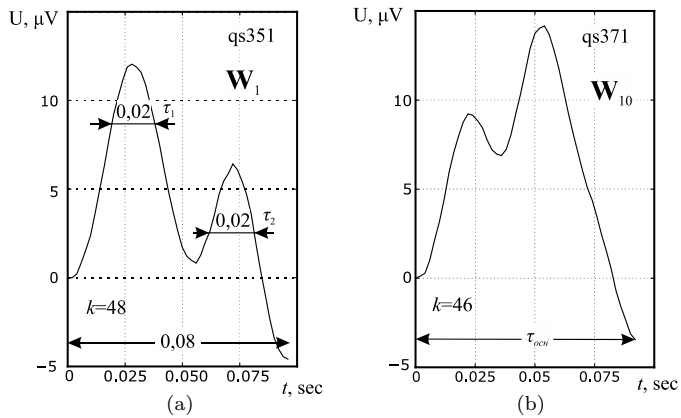


Figure 16.17. Segments of the EEG helmet wave: (a) $\mathbf{W}_1$; (b) $\mathbf{W}_{10}$.

16.9 Secondary Processing of the Epochs Segments

16.9.1 *SPU setup*

At this stage of EEG analysis, a classification decision is taken regarding the patterns formed by segments $\mathbf{W}_r = \{\gamma_r(n)\}_0^{k_r-1}$. Here, k_r is the dimensionality and $\gamma_r(n)$ is the EV of the segment envelope contour. It is assumed that segments $\mathbf{W}_r, r = 0, 1, \ldots, R-1$, were classified at the stage of primary processing of the EEG epoch segments. The pattern $\mathbf{Q}_v = \{\mathbf{W}_{r_v}\}_{r_v}^{k_v+s_v-1}$ consists of $\mathbf{s}_v$ segments $\mathbf{W}_{r_v}$, $\mathbf{W}_{r_v+1}, \cdots, \mathbf{W}_{r_v+s_v-1}$ following one another and numbered in sequence. As noted earlier, as pattern examples we used segments $\mathbf{W}_0$, $\mathbf{W}_1$, $\mathbf{W}_2$ (Figure 16.5) that make up the pathological pattern "slow wave, spike, sharp wave", segments $\mathbf{W}_{33} \div \mathbf{W}_{36}$ (Figure 16.5) constituting the pattern "multiple spikes, slow wave" (Figure 16.5), and segments $11 \div 19$ (Figure 16.5) composing the 'Spindle' pattern as a chain of amplitude-modulated sinusoidal oscillations, typical of the EEG of a healthy person during the awake state.

The 'Spindle' pattern in its undistorted form can be observed following band-pass filtering in the α-band (Figure 16.18). The basic secondary processing procedure which is considered in this chapter is the statistical analysis of the informative features presented in Table 16.2 and obtained during the primary processing. The EEG 'Autumn-2016' is used as an example.

16.9.2 *Distribution of the dimensionalities of segment groups*

The dimensionality k_r of the segment $\mathbf{W}_r$ is equal to the number of EVs $\gamma(n) = t_\partial + i\Delta u(n)$ in its complex envelope. The EEG wavelength described by the envelope is equal to $\tau_r = \kappa_r t_\partial = 0,002\,\kappa_r$ s. The sampling $\mathbf{K}_{j,l} = \{\kappa_r\}_j^l$ is a sequence of dimensionality values of the segments $\mathbf{W}_j$, $\mathbf{W}_{j+l}, \ldots \mathbf{W}_l$.

Assuming the law of the distribution of probabilities $P_{j,k}(\kappa_r)$ of the sample values $\mathbf{K}_{jl}$ to be normal, we provide its characterization using the mathematical expectation parameters $\bar{\kappa}$ and the CKO σ_κ.

The sample of the values κ_r, $r = 0, 1, \ldots, 21$, is given in line 2 of Table 16.2. The required parameters of distribution $P_{0,21}(\kappa_r)$ are

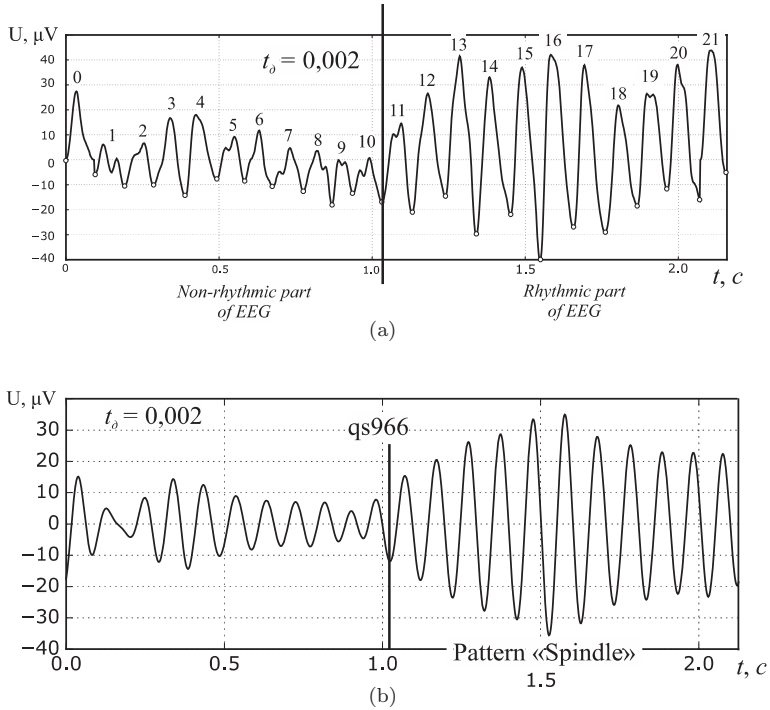


Figure 16.18. Types of EEG 'Autumn 2016' (a) at the input of SPU; (b) at the output of the filter with the band $(8 \div 14)$ Hz in the α band.

equal to

$$\bar{\kappa}_{0,21} = 48, \quad \sigma_{\kappa,0,21} = 4.66, \quad \sigma^2_{\kappa,0,21} = 21.72. \tag{16.23}$$

Let us test the null hypothesis on the significance of differences between the general means $\bar{\kappa}_{0,10}$ and $\bar{\kappa}_{11,21}$ of the distributions $\mathbf{P}_{0,10}(\kappa_r)$ and $\mathbf{P}_{11,21}(\kappa_r)$ obtained by dividing the sample $\mathbf{K}_{0,21}$ into two samples. The first sample $\bar{\kappa}_{0,10}$ is composed of the dimensionalities of segments $\mathbf{W}_0 \div \mathbf{W}_{10}$, and the second sample consists of $\mathbf{W}_{11} \div \mathbf{W}_{21}$ dimensionalities. Statistical parameters of these samples are as follows:

$$\begin{aligned} &\bar{\kappa}_{0,10} = 45.91, \quad \sigma_{\kappa_{0,10}} = 4.74; \quad \sigma^2_{\kappa_{0,10}} = 22.49; \quad n_1 = 11; \\ &\bar{\kappa}_{11,21} = 50.09; \quad \sigma_{\kappa_{11,21}} = 3.67; \quad \sigma_{\kappa_{0,10}} = 4.74; \quad n_2 = 11. \end{aligned} \tag{16.24}$$

Here, n_1 and n_2 indicate the number of elements in each sample. The ratio $|\bar{\kappa}_{0,10} - \bar{\kappa}_{11,21}|$ to its statistical error [14] was selected as a

Table 16.2. Informative features of the EEG 'Autumn 2016' segments available for the secondary processing (fragment).

Segment $\mathbf{W}_r$	$\mathbf{W}_0$	$\mathbf{W}_1$	$\mathbf{W}_2$	⋯	$\mathbf{W}_9$	$\mathbf{W}_{10}$	$\mathbf{W}_{11}$	$\mathbf{W}_{12}$		$\mathbf{W}_{19}$	$\mathbf{W}_{20}$	$\mathbf{W}_{21}$
Dimensionality, k_r	46	48	47		33	46	49	52		47	53	42
Peak-to-peak amplitude, μV	27.5	16.6	17.2		17.8	16.6	35.53	47.6		45	54	49
Energy, mV^2	0.008	0.001	0.0015		0.002	0.003	0.007	0.013		0.015	0.023	0.03
δ	0.003	0	0	⋯	0.003	0.001	0	0.001	⋯	0.003	0.004	0.009
θ	0.003	0	0		0	0	0.001	0.004		0.005	0.009	0.011
α	0.002	0.0001	0.0006		0	0.0005	0.003	0.005		0.005	0.008	0.008
β_1	0.007	0.004	0.004		0.0 006	0.0006	0.002	0.001		0.001	0.001	0.002
β_2	0	0.002	0		0.0003	0	0.0006	0		0.007	0	0
Frequency band	θ, δ, α	β_1, α	α, β_1		β_1, β_2	δ, β_1, β_2	α, β	α, β		θ, α	θ, α	θ, δ
Dominant frequency of the segment	10.9	10.4	10.7		15.2	10.9	10.2	9.6		10.6	9.4	11.9
Sampling frequency	500	500	500		500	500	500	500		500	500	500 500
Measure of similarity with a complex sin/cos	0.84	0.92	0.312		0.78	0.66	0.84	0.945		0.845	0.854	0.766
Energy fraction in the α band	0.23	0.19	0.42		0.029	0.15	0.47	0.43		0.31	0.35	0.26
Indices of vertices	20	12.32	31		9.20	10.26	18.31	24			16	20
τ at the base level	0.09	0.1	0.09		0.07	0.08	0.09	0.11		0.09	0.1	0.08
At 0.707	0.02	0.02/0.02	0.02		0.04	0.02	0.05	0.01		0.05	0.04	0.04
Vertex angles	36°	27°, 27°	86°		38°, 42°	40°, 25°	21°, 46°	31°		84°	15°	31°
Symmetry coefficient	0.67	0.26	0.72		0.66	0.7	0.83	0.92		0.81	0.79	0.93

fidelity criterion for estimating the significance of differences between the general means $\bar{\kappa}_{0,10}$ and $\bar{\kappa}_{11,21}$. To make a decision, values t_ϕ of the actual value of this criterion are compared with the standard value $t_{\rm st}$ of the normalized deviation. The quantitative value of t_ϕ is calculated as follows [14]:

$$t_\phi = \frac{|\bar{\kappa}_{0,10} - \bar{\kappa}_{11,21}|}{\sqrt{\frac{(n_1-1)\sigma^2_{0,10}+(n_2-1)\sigma^2_{11,21}}{n_1+n_2-2} \cdot \frac{n_1+n_2}{n_1 \cdot n_2}}}. \tag{16.25}$$

Substituting the value from (16.24) into this expression yields $t_\phi = 2.317$. Standard t-test values for the number of degrees of freedom $n_1 + n_2 - 2 = 20$ are $t_{\rm st} = 2.85$ for the significance level $P_{3H} = 0.01$ [14]. Comparing the values t_ϕ and $t_{\rm st}$ obtained leads to the conclusion that the *null hypothesis is valid*. This is in agreement with the statement that the general means of the distributions $P_{0,10}(\kappa_r)$ and $P_{11,21}(\kappa_r)$ coincide, and the differences between calculated parameters $\bar{\kappa}_{0,10}$ and $\bar{\kappa}_{11,21}$ were caused by some accidental but not systemic reasons.

16.9.3 *Distribution $P(A_r)$ of the EEG segment peak-to-peak amplitudes*

The sample of values $\{A_r\}_0^{21}$ is presented in line 2 of Table 16.2 and is characterized by the following mathematical expectation values: $\bar{\mathbf{A}} = 40.7\,\mu\text{V}$, MSD $\sigma = 21.7\,\mu\text{V}$, the variance of $\sigma^2 = 471\,\mu\text{V}^2$, and the number of sample components $n = 22$. We divide the sample $\{A_r\}_0^{21}$ into two samples $\mathbf{A}_{0,10}$ and $\mathbf{A}_{11,21}$ of the same dimensionality $n_1 = n_2 = 11$. Then, we test the null hypothesis in relation to these samples using the expression (16.25) with the following values contained in it:

$$m_{A_{0,10}} = 21.73; \quad \sigma_{A_{0,10}} = 5.87; \quad \sigma^2_{A_{0,10}} = 31.2;$$

$$m_{A_{11,21}} = 58.44; \quad \sigma_{A_{11,21}} = 14.66; \quad \sigma^2_{A_{11,21}} = 214.9.$$

As a result, the value of the criterion $t_\phi = 7.71$, and the standard t-test values for the number of degrees of freedom $k = 20$ for $P = 0.01$ is equal to 2.85. Given that $t_\phi > t_{\rm st}$, then the null hypothesis for the peak-to-peak amplitudes of the potentials of segments $\mathbf{W}_0 \div \mathbf{W}_{21}$ has not proved to be true.

We approach the question on which sample elements $\{A_r\}_0^{21}$ have different probability distribution laws from the perspective of the theory of detecting a *disorder* r_{00} *in* a random process [2]. We divide the sample into subgroups G_l of same dimensionality s. The value s is selected on the basis of a compromise between the sufficient representativeness of the group and the number of subgroups. With $s = 5$, MSD of the peak-to-peak values within the subgroup decreases by 2.2 times, and the number of individual subgroups composed of 22 segments is equal to 18. A separate lth subgroup is indicated as $G_l = \{A_r\}_l^{l+1}$, $l = 0, 1, \ldots, 17$. For instance, G_4 contains the peak-to-peak amplitudes of the segments $\{W_r\}_4^8$. The fidelity criterion for the differences between the general means $\bar{m}_l$ and $\bar{m}_{l+5}$ of the peak-to-peak sample subgroups is the ratio of the difference $|\bar{m}_l - \bar{m}_{l+5}|$ to its statistical error $\sqrt{\sigma_l^2 + \sigma_{l+5}^2}$ [14], i.e. with the identical dimensionalities $s = 5$ of the subgroups and $k = 5$, we get

$$t_\phi = \frac{|\bar{m}_l - \bar{m}_{l+5}|}{\sqrt{\sigma_l^2 + \sigma_{l+5}^2}}\sqrt{n} > t_{\text{st}} \; npu \; k = 8. \tag{16.26}$$

The results of the parameter calculation for accepting the null hypothesis regarding the peak-to-peak sample subgroups were obtained using this formula and presented in Table 16.3. The latter shows that the expression $t_\phi > t_{\text{st}}$ is valid for the subgroups G_5 and G_{11}, comprising the segments $\{\mathbf{W}_r\}_0^{10}$ and $\{\mathbf{W}_r\}_{11}^{15}$, respectively.

Based on this parameter, we single out segments $\{\mathbf{W}_r\}_0^{10}$ into a separate group. Hence, the first point of disorder for the general mean values in the distribution of swings regarding the EEG 'Autumn 2016' is the point $l_{00} = 10$, which is related to the boundary of the subgroups G_6 and G_{11}, consisting of the segments $\{\mathbf{W}_r\}_6^{10}$ and $\{\mathbf{W}_r\}_{11}^{15}$, respectively (Figure 16.18(a)).

Now, we verify the correctness of the null hypothesis relating to the general mean for segments $\mathbf{W}_{11}, \ldots \mathbf{W}_{20}$, which were not included in the first group $\{\mathbf{W}_r\}_0^{10}$. To this end, the segments of the second group $\{\mathbf{W}_r\}_0^{20}$ are divided into two equal-number subsets of elements $\{\mathbf{W}_r\}_{11}^{15}$ and $\{\mathbf{W}_r\}_{16}^{21}$, and their general means are estimated. The evaluation criterion is the standard value of the normalized deviation t_{st}, which the actual value t_ϕ of this criterion is compared with. The values $t_\phi = 0.09$ and $t_{\text{st}} = 3.36$ at a significance level $P = 0.01$.serve

Table 16.3. Results of calculation of the null hypothesis credibility for the peak-to-peak sample subgroups with $s = 5$.

	Compared subgroups				
1	$\mathbf{G_1}$	$\mathbf{G_{1+5}}$	t_f	t_{st}	
0	$\mathbf{G_0} = \{r0, r1, r2, r3, r4\}$	$\mathbf{G_5} = \{r_5, r_6, r_7, r_8, r_9\}$	1.8	3.36	+
1	$\mathbf{G_1} = \{r_1, r_2, r_3, r_4, r_5\}$	$\mathbf{G_6} = \{r_6, r_7, r_8, r_9, r_{10}\}$	0.91	3.36	+
2	$\mathbf{G_2} = \{r_2, r_3, r_4, r_5, r_6\}$	$\mathbf{G_7} = \{r_7, r_8, r_9, r_{10}, r_{11}\}$	0.53	3.36	+
3	$\mathbf{G_3} = \{r_3, r_4, r_5, r_6, r_7\}$	$\mathbf{G_8} = \{r_8, r_9, r_{10}, r_{11}, r_{12}\}$	0.5	3.36	+
4	$\mathbf{G_4} = \{r_4, r_5, r_6, r_7, r_8\}$	$\mathbf{G_9} = \{r_9, r_{10}, r_{11}, r_{12}, r_{13}\}$	1.46	3.36	+
5	$\mathbf{G_5} = \{r_5, r_6, r_7, r_8, r_9\}$	$\mathbf{G_{10}} = \{r_{10}, r_{11}, r_{12}, r_{13}, r_{14}\}$	2. 78	3.36	+
6	$\mathbf{G_6} = \{r_6, r_7, r_8, r_9, r_{10}\}$	$\mathbf{G_{11}} = \{r_{11}, r_{12}, r_{13}, r_{14}, r_{15}\}$	5.05	3.36	−
7	$\mathbf{G_7} = \{r_7, r_8, r_9, r_{10}, r_{11}$	$\mathbf{G_{12}}\} = \{r_{12}, r_{13}, r_{14}, r_{15}, r_{16}\}$	6.65	3.36	−
8	$\mathbf{G_8} = \{r_8, r_9, r_{10}, r_{11}, r_{12}$	$\mathbf{G_{13}}\} = \{r_{13}, r_{14}, r_{15}, r_{16}, r_{17}\}$	6.4	3.36	−
9	$\mathbf{G_9} = \{r_9, r_{10}, r_{11}, r_{12}, r_{13}\}$	$\mathbf{G_{14}} = \{r_{14}, r_{15}, r_{16}, r_{17}, r_{18}\}$	2.6	3.36	+
10	$\mathbf{G_{10}} = \{r_{10}, r_{11}, r_{12}, r_{13}, r_{14}\}$	$\mathbf{G_{15}} = \{r_{15}, r_{16}, r_{17}, r_{18}, r_{19}\}$	1.45	3.36	+
11	$\mathbf{G_{11}} = \{r_{11}, r_{12}, r_{13}, r_{14}, r_{15}\}$	$\mathbf{G_{16}} = \{r_{16}, r_{17}, r_{18}, r_{19}, r_{20}\}$	0.22	3.36	+



Figure 16.19. Classes (patterns) of the EEG 'Autumn 2016' with different values of the general means in regard to the segment peak-to-peak magnitudes.

as the results of calculations. Due to the fact that $t_\phi \ll t_{st}$, the null hypothesis is true, i.e. the general means for the subsets $\{\mathbf{W}_r\}_{11}^{15}$ and $\{\mathbf{W}_r\}_{16}^{20}$ do not differ from each other.

Thus, according to the general mean values, the sample of the segment peak-to-peak amplitudes in the EEG 'Autumn 2016' is partitioned into two classes (patterns) ($\mathbf{Q}_1 = \{\mathbf{W}_r\}_0^{10}$ and $\mathbf{Q}_2 = \{\mathbf{W}_r\}_{11}^{20}$) (Figure 16.19), whose sample peak-to-peak values are as follows:

$$\mathbf{Q}_1 = \{A_r\}_0^{10} = \{27; 17; 17; 31; 32; 19; 22; 18; 21; 18; 36\};$$

$$\mathbf{Q}_2 = \{A_r\}_{11}^{20} = \{36; 48; 74; 63; 77; 19; 82; 18; 67; 51; 45; 54\}.$$

The parameters of the peak-to-peak amplitudes of pattern segments $\mathbf{Q}_1$ and $\mathbf{Q}_2$ are equal to $\bar{A}_1 = 21,73$; $\sigma_{A_1} = 5.86$; $\bar{A}_2 = 58.41$; $\sigma_{A_2} = 14.66$. The statistical analysis of the potential peak-to-peak values of the EEG 'Autumn 2016' segments showed that this EEG consists of patterns $\mathbf{Q}_1$ and $\mathbf{Q}_2$. Therefore, a subsequent analysis of the informative features of these patterns will be carried out in order to associate them with known EEG patterns.

16.9.4 *Distribution of the values of the segment energy E_r*

Let us consider the null hypothesis related to the general mean energy values in the classes $\mathbf{Q}_1$ and $\mathbf{Q}_2$. The mathematical expectations E_1 and E_2 of the energy values with MSD σ_{E_1} and σ_{E_2} in the segments of these classes are $\bar{E}_1 = 3128\,\mu\text{V}^2$; $\bar{E}_2 = 22436\,\mu\text{V}^2$; $\sigma_{E_1} = 2376\,\mu\text{V}^2$; $\sigma_{E_2} = 2376\,\mu\text{V}^2$. In accordance with the expression (26) for $k = 20$ degrees of freedom and $n = 22$, we obtain the actual value of the criterion $t_\phi = 6.27$. It significantly exceeds the standard t-test value $t_{\text{st}} = 3.85$ [14] for $k = 20$ degrees of freedom. Therefore, the null hypothesis on the equality of the mean values in the patterns $\mathbf{Q}_1$ and $\mathbf{Q}_2$ is rejected, and the probability distributions of the energy values in these patterns are systemically different from each other. We arrive at similar conclusions in regard to the probability distribution of the symmetry coefficients of these classes.

16.9.5 *Distribution of the measures of similarity of the segment envelopes with a complex sinusoid*

Line 7 of Table 16.2 contains the values of the measures of shape similarity ε_r of the envelope of the EEG 'Autumn 2016' current segment $\mathbf{W}_r$ with a sinusoidal oscillation with a frequency $f_1 = 1/\kappa_r, r = 0, 1, \ldots, 21$. To eliminate the effect of random phase shifts between the segment envelope within the phases 270°–630° (Figure 16.8), the value ε was computed as a sum of the envelope measures of similarity with the fragments of a sinusoid and a cosine that have the same period. Parameters of the mathematical expectation $\bar{\varepsilon}$ and variance

$\sigma^2_{\bar{\varepsilon}}$ for the entire sample of the segments $\{\mathbf{W}_r\}_0^{21}$ are equal to $\bar{\varepsilon} = 0.8$; $\sigma^2_{\varepsilon} = 0.019$.

Let us test the credibility of the null hypothesis of the random variable under consideration for the segments of classes $\mathbf{Q}_1 = \{\mathbf{W}_r\}_0^{10}$ and $\mathbf{Q}_2 = \{\mathbf{W}_r\}_{11}^{21}$. The mathematical expectation values $\bar{\varepsilon}_1$ and $\bar{\varepsilon}_2$, as well as the variance values $\sigma^2_{1,\varepsilon}$ and $\sigma^2_{2,\varepsilon}$ are expressed as $\bar{\varepsilon}_1 = 0.73; \sigma^2_{1,\varepsilon} = 0.027; \bar{\varepsilon}_2 = 0.873; \sigma^2_{2,\varepsilon} = 0.002$. The calculation of the actual criterion value in accordance with (16.26), when $n_1 = n_2 = 11$ yields $t_\phi = 3.174$, which exceeds the standard t-test value $t_{\text{st}} = 2.85$ for the number of degrees of freedom $k = 20$ at the significance level $P = 0.01$. Therefore, the null hypothesis is rejected and, with a 0.99 probability, segments of the pattern $\mathbf{Q}_2$ exhibit the measure of shape similarity of their envelope which is much closer to $\varepsilon = 1$, than to the segment of pattern $\mathbf{Q}$ (Figure 16.9).

16.9.6 *The energy of the segments of classes* $\mathbf{Q}_1$ *and* $\mathbf{Q}_2$ *in the informative frequency bands*

Figure 16.20 presents diagrams of the distribution of the averaged energy fractions of the segments of classes $\mathbf{W}^{(1)}$ and $\mathbf{W}^{(2)}$ across the informative frequency bands $\delta, \theta, \alpha, \beta_1, \beta_2$; the diagrams were made using the data from Table 16.2. Each diagram consists of five counts shaped as rectangles whose positions correspond to particular frequency bands, with the height of each rectangle indicating the averaged fraction (%) of the energy of all segments belonging to the class and present in this band.

An individual diagram contains five rectangular-shaped samples whose positions correspond to particular frequency bands and heights correspond to the averaged fraction (%) of the total energy of all segments of the pattern in the bands. The diagrams suggest the following:

(1) The maximum amount of the segment energy of the patterns $\mathbf{Q}_1$ and $\mathbf{Q}_2$ is concentrated in the α-band.
(2) The segment fractional energy of the pattern $\mathbf{Q}_1$ is distributed relatively evenly across the remaining bands (with the exception of the β_1 band with its energy level of 27%). Such a broadband

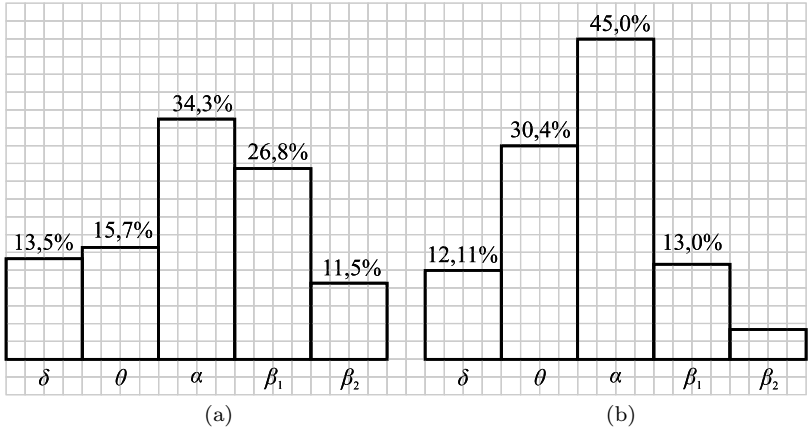


Figure 16.20. Diagrams of the segment energy distribution related to (a) class $\mathbf{Q}_1$; (b) class $\mathbf{Q}_2$.

nature of the spectra of segment envelopes leads to a wide variety of segment shapes, as well as a lack of a single rhythm due to the slight dominance of the energy in the α band.

(3) About 70% of the segment energy of the pattern $\mathbf{Q}_2$ corresponds to the α band and the adjacent θ-band. The segment energy in the other bands does not exceed 30% of the pattern's total energy. Therefore, the oscillation frequency of approximately 10 Hz dominates in the bands, in which the corresponding rhythm is observed.

The findings obtained correlate well with the data presented in Figure 16.18. Pattern $\mathbf{Q}_1$ will be hereinafter referred to as the arrhythmic part of the EEG 'Autumn 2016', and pattern $\mathbf{Q}_2$ as the rhythmic part of the signal.

16.9.7 *Classification of patterns based on the results of the secondary processing of the EEG 'Autumn 2016' segments*

During the secondary processing of the EEG, two patterns (classes and sets) of segments $\mathbf{Q}_1$ and $\mathbf{Q}_2$. were identified. The informative features of the segments within a separate pattern have common distinctive values. The following informative features of the segments

were analyzed in the process of secondary processing,: dimensionality κ_r, peak-to-peak amplitudes of potentials A_r, values of energy E_r symmetry coefficients, measures of similarity of the segment envelope with a complex sinusoid, and distribution of the segment energy fractions across the informative frequency ranges δ, θ, α, β_1, and β_2. Specific values of the features were calculated as random variables with the probability distributions approaching the normal law. The detection of the pattern limits in the analysis of an individual feature was associated with solving the problem of random process disorder. It was assumed that the pattern segments are characterized by the general means that systematically differ from the general means outside the pattern. The disorder was detected in the analysis of the peak-to-peak values of the segment potentials, and they were split into the two classes $\mathbf{Q}_1 = \{\mathbf{W}_r\}_0^{10}$ and $\mathbf{Q}_2 = \{\mathbf{W}_r\}_{11}^{21}$. Subsequent statistical analysis of the segment informative features confirmed the correctness of this decision with a probability of at least 0.99.

In a normal EEG of a healthy individual, spontaneous changes in the amplitude of the α-rhythm are regularly observed; the changes are expressed in a gradual increase and a gradual decrease in the amplitude which forms the characteristic 'Spindle' pattern (Figure 16.18(b)). In the pattern $\mathbf{Q}_2$, the α-rhythm dominates, and the 'Spindle' pattern is prominent, which consists of the segments with a peak-to-peak amplitude of $35 \div 82\,\mu\text{V}$ and has a stable frequency $(9,1 \leq f \leq 10,6)$ Hz. The segments $\mathbf{W}_{13}, \mathbf{W}_{14}, \mathbf{W}_{15}, \mathbf{W}_{17}, \mathbf{W}_{18}$, and $\mathbf{W}_{20}$ have pointed tops. Due to the presence of such a large number of peaked waves, the pattern $\mathbf{Q}_2$ was classified as a *boundary* EEG [12].

16.9.8 *The results of the secondary processing of the EEG 'Autumn 2016'*

The array of segments $\{\mathbf{W}_r\}_0^{21}$, composing the EEG 'Autumn 2016' has been reasonably divided into two classes: $\mathbf{Q}_1 = \{\mathbf{W}_r\}_0^{10}-$ the arrhythmic component of the oscillation and $\mathbf{Q}_2 = \{\mathbf{W}_r\}_{11}^{21}$ as its rhythmical component. The arrhythmic component makes up the pattern "Desynchronization of the neuron activity". The EEG rhythmic component is represented by the 'Spindle' pattern. During the formation of the first pattern, increased functional activity of the brain was observed in the patient. Over time, the brain activity reached a balanced state, in which the 'Spindle' pattern was formed.

The segments of the EEG 'Autumn-2016' possess a number of features based on which the oscillation under analysis was classified as *a boundary* EEG [17].

16.10 Conclusions

1. Solving the problem of evaluating the state of the central nervous system is related to the analysis of electric oscillations generated by the brain. The oscillations are recorded as the EEG with a system of electrodes placed on the scalp surface. CT, MRI, PET, and other imaging methods are widely used in neurology for diagnosing the state of the brain structures by means of the visualization of its internal structures. The conceptual difference between the methods is as follows: clinical EEG ensures the diagnosis of the CNS *functional activity*, while the neurovisualization methods allow for the *diagnosis of organic* brain lesions.
2. For the optimal processing of the EEG signals according to the quality criteria specified, the statistical characteristics of the signals and noises must be known. Currently, the information is not available to neurophysiologists, and, therefore, the issue of the optimum EEG processing remains open.

   The EEG is considered to be a superposition of weakly correlated signals from the sources located in the cerebral cortex. Other sources are of lower activity and located in the deep brain structures. The author's abstract of V.A. Ponomarev's doctoral dissertation (scientific adviser: Yu. D. Kropotov, Prof., Dr. Biol. Sc.) contains the most recent data on the EEG genesis [15]. Local areas of the brain — groups of compactly arranged and synchronized pyramidal neurons whose superposition is described by the instantaneous mixing linear model — are taken as elementary sources. The quantities measured $x(t)$, i.e. EEG is a linear combination of signals from the field sources $s(t) : x_i(t) = \sum_{j=1}^{M} a_{i,j} s_j(t) = As(t)$. Here, $a_{i,j}$ is an element of the matrix A, characterizing the impact of the jth source on the potential of the ith electrode. The signals $s_j(t)$ are not measured directly and are "concealed" from monitoring. As noted in [15], it is possible to broadly estimate the elements of matrix A using the physical principles of field propagation and knowledge of brain anatomy and by solving the inverse problem for the field source localization [10].

The review given above explains the complexity of obtaining the signal $x(t)$ analytically. This circumstance has resulted in the current absence of algorithms for EEG signal processing that would ensure obtaining the results with the maximum signal-to-noise ratio and with the maximum probability of sound classification for a specified error probability, etc.

3. A structural (syntactic) approach has been proposed to EEG analysis interpreting the process of the visual analysis of the signal by the clinician. The EEG is considered to be a set of simple subimages (segments); EEG waves were taken as the segments. The shape of a segment, its duration, and peak-to-peak amplitude, as well as its time and frequency parameters, serve as the informative features of the EEG fragment specifying the segment. Each segment is characterized by its location; the sum of all segments makes up the original EEG, and the spectrum of the entire EEG is equal to the sum of the spectra of all its segments. This presentation of the EEG ensures not only the time and frequency resolution of the signal but also allows estimating the contribution of an individual segment to the differential and integral characteristics of the EEG on the whole.
4. Mathematical apparatus for EEG analysis has been developed based on the structural approach perspective. The EEG is interpreted as a contour of an image in the form of a sequence of elementary vectors interpolating the adjacent samples from the electroencephalograph output. Contour analysis is part of the theory of signals set in a vector linear space. This is a unitary space where the scalar product of vectors is defined, and the vectors are presented in a complex-valued form. Assigning signals in a unitary space has several advantages, among which we should note the following: it ensures an unambiguous determination of the angle between vectors; the measure of similarity between the signals is invariant to the magnitude of the angle between vectors, the scalar product between the vectors is more informative than vector assignment in a real linear space [3].

Due to these properties of a unitary space, contour analysis provides a unified approach to all operations related to the processing of a structurally assigned EEG: segmentation, generation of time and frequency informative features of the segments, EEG classification into the normal, borderline, and pathological classes.

It is essential to note that the structural approach to presenting the EEG in combination with presenting its contour as an element of a unitary space ensures the following advantages of the analysis conducted:

1. All the EEG count operations are *performed without human intervention*, which enables the *automatic analysis* of the EEG.
2. The EEG analysis system based on the approaches described above can be interpreted as a clinician *support system*.

References

[1] Basswil, M. *et al.* (1989) *Detection of Changes in the Properties of Signals and Dynamical Systems* (Mir, Moscow) (in Russian).

[2] Bodenstein, G. and Praetorius H.M. (1977) Feature extraction from the electroencephalogram by adaptive segmentation. *Proceedings of IEEE*, 65, 642–652.

[3] Efimov, N.V. and Rozendorn E.R. (1974) *Linear Algebra and Multidimensional Geometry* (Gl. Red. Fiz.-Mat. Lit. Nauka, Moscow) (in Russian).

[4] Fu, K.S. (1977) *Syntactic Methods in Pattern Recognition* (Mir, Moscow) (in Russian).

[5] Furman, Y.A., Egoshina, I.L., and Ivanov, K.O. (2016) Analysis and classification of the elements of electroencephalogram on the basis of its contour model. *Proceedings of Republic Scientific Seminar "Methods of Modeling"*, Kazan, pp. 52–93 (in Russian).

[6] Furman, Y.A., Krevetskii, A.V., Peredreev, A.K., Rozhentsov, A.A., Khafizov, R.G., Egoshina, I.L., and Leukhin, A.N. (2002) *Introduction to Contour Analysis and Its Application in Image and Signal Processing* (Fizmatlit, Moscow) (in Russian).

[7] Furman, Y.A., Sevastyanov, V.V., Ivanov, K.O., and Kazarinov, A.V. (2015) The way to segment the fine structure in electroencephalogram. *Vestn. Ryazansk. Gos. Radiotekhn. Univ.*, 54(2), 56–67.

[8] Furman, Y.A., Sevastyanov, V.V., and Ivanov, K.O. (2016) Contour analysis of a fine structure in an electroencephalogram. *Pattern Recognition Image Analysis*, 26, 758–772.

[9] Furman, Y.A., Sevastyanov, V.V., Ivanov, K.O., Khafizov, R.G., Svintsov, S.N., Kazarinov, A.V., and Ohotnikov, S.A. (2015) Contour mathematical model of EEG. *Vestn. Povolzhsk. Gos. Tekhnol. Univ. Ser. Radiotekhn. Infokommun. Sist.*, 2(26), 50–61.

[10] Gnezditskii, V.V. (2004) *The Inverse Problem of EEG and Clinical Electroencephalography* (MEDpress-inform, Moscow) (in Russian).
[11] Kaplan, A.Y. (1999) The problem of segmental description of the human electroencephalogram. *Fiziologiia Cheloveka*, 25(1), 125–133.
[12] Kropotov, Y.D. (2010) *Quantitative EEG, Cognitive Evoked Potentials of the Human Brain, and Neurotherapy* (Zaslavsky AYu, Donetsk) (in Russian).
[13] Kulaichev, A.P. (2002) *Computer Electrophysiology* (MGU, Moscow) (in Russian).
[14] Lakin, G.F. (1980) *Biometrics* (Vysshaya Shkola Publishers, Moscow) (in Russian).
[15] Ponomarev, V.A. (2016) Hidden sources of the EEG and event-related potentials, and their significance: Department of Biological Science dissertation, St. Petersburg (in Russian).
[16] Rusinov, V.S., Grindel, O.M., Boldyreva, G.N., and Vakar, E.M. (1987) Biopotentials of the human brain. *Mathematical Analysis* (Meditsina, Moscow) (in Russian).
[17] Tsygan, V.N., Bogoslovskii, M.M., and Mirolubov, A.V. (2008) *Electroencephalography.* D'yakonov, M.M. (ed.) (St. Petersburg, Nauka) (in Russian).
[18] Zenkov, L.R. (2013) *Clinical Electroencephalography* (MEDpress-inform, Moscow) (in Russian).
[19] Zhadin, M.N. (1984) *Biophysical Mechanisms of Electroencephalogram Formation* (Nauka, Moscow) (in Russian).

© 2025 World Scientific Publishing Company
https://doi.org/10.1142/9789811267215_0017

Chapter 17

The Early Stages of Post-traumatic Epileptiform Activity and Parkinson's Disease Based on EEG Wavelet Spectra Features

Yury V. Obukhov[*,‡], **Konstantin Yu. Obukhov**[*,§], **Ivan A. Kershner**[*,||], **and Ilya G. Komoltsev**[†,¶]

[*]*Kotel'nikov Institute of Radioengineering and Electronics of Russian Academy of Sciences, Moscow, the Russian Federation*
[†]*Institute of Higher Nervous Activity and Neurophysiology of Russian Academy of Sciences, Moscow, the Russian Federation*
[‡]*yuvobukhov@mail.ru*
[§]*ko.arsenalfc@gmail.com*
[||]*ivan_kershner@mail.ru*
[¶]*outaudiofillin@gmail.com*

The approach and results of extraction EEG wavelet spectra features and classification in this features space the early stages of post-traumatic epileptiform activity in experimental rats after brain injury, and Parkinson's disease are described. Feature extraction method is based on wavelet spectrograms ridges, and local extrema points time–frequency distribution. Proposed methods and algorithms are used for post-traumatic epileptiform activity recognition in long durable EEG rat records before and after traumatic brain injury. Feature extraction and classification model of the early-stage Parkinson's disease in EEG feature space can be applicable to disease risk group identification and screening.

Keywords: Post-traumatic epileptiform activity, epileptiform activity detection, electroencephalography, EEG, spindles detection, preclinical diagnosis, wavelet spectrogram, seizure detection.

17.1 Introduction

Wavelet analysis of non-stationary signals processing is a well -known and powerful technique in various areas of applications [16], so as in neuroscience [11], electroencephalography (EEG) data analysis [15, 27,28], and brain pathology investigations and diagnostics, especially epileptiform activity detection and prediction in EEG [1–4, 6, 7, 9, 14, 21, 22, 24, 28] and sleep spindles detection [29].

Here we describe the approach and results of extraction EEG wavelet spectra features and classification in this feature space the early stages post-traumatic epileptiform activity in experimental rats after brain injury [19, 20] and Parkinson's disease [18, 25].

It is well known that Traumatic Brain Injury (TBI) develops post-traumatic epilepsy (PTE) in a significant number of patients (approximately 20%). We consider the algorithms of automatic detection and classification of the epileptiform discharges (ED) and the sleep spindles (SS). Detection and classification of the early-stage ED up to 10 days after TBI in the presence of SS is a difficult problem because both have the same etiology in this period.

Parkinson's disease (PD) belongs to a wide class of neurodegenerative diseases and is caused by the death of dopaminergic neurons of the brain. Attention was paid to the mechanisms of brain plasticity, which serve to compensate functional insufficiency of the degenerating neurons. From this point of view, the authors consider the dynamic of neurodegenerative diseases and state the necessity of preclinical diagnostics and preventive therapy development. The main problem of PD diagnostics is the search for disease features at pre-clinical and early clinical stages [26].

Electroencephalography (EEG) and magnetoencephalography (MEG) are applicable methods for brain electrical activity analysis of PD patients [5, 23]. Disorders of different organism systems, such as movement disorders, vegetative, emotional, or physical, are considered as clinical features of PD. It is assumed that such disorders reflect or are coursed by brain electrical activity. This chapter

proposes an approach for feature extraction and classification of the early-stage PD in EEG feature space [24]. This model can be applicable to disease risk group identification and screening. The investigated groups consisted of 31 non-treated PD patients with clinical diagnosis of 1st stage according to Hoehn–Yahr scale [15] and 18 control volunteers. Prediction results were tested on different sets of 22 PD patients and 16 normal volunteers.

17.2 Detection and Classification of Post-Traumatic Epileptiform Activity in EEG of Rats after Traumatic Brain Injury

17.2.1 *Experiment design*

The main experiment was conducted on adult male 24-month-old Wistar rats. The animal was kept in a home cage with free access to food and water. Four stainless steel screw electrodes were implanted bilaterally under the somatosensory cortex (1 mm rostral and 3 mm lateral from bregma) and under the visual cortex (7 caudal and 3 mm lateral from bregma), while referential and ground electrodes were located in the occipital bone near the midline. Electrodes were fixed with dental cement.

In 7 days after electrodes implantation, the lateral fluid percussion (LFP) injury model was used to reproduce TBI [19]. Craniotomy ($d = 3$ mm) was drilled in the temporary bone (3 mm to caudal and 3 mm lateral from bregma), and the injury with pressure of 3.4 atm was performed. Rat was placed in the home cage immediately after injury. EEG was measured via Bio Recorder (24-bit analog front-end for the measurement of biopotentials) for 14 days: 7 days prior to TBI and 7 days after TBI in acute post-traumatic period.

Three experts in neurobiology performed a blind manual markup of the EEG records into SS and ED. The first expert examined EEG records and selected suitable ED with 2–3 s background rhythm fragments before and after ED. High-voltage rhythmic spikes (HVRS), or spike-wave-discharges (SWD), were selected as ED. Recognition was based on certain parameters: sudden or rapid onset with high amplitude spikes, rhythmicity of spikes (relatively stable interval between

spikes), high voltage of spikes compared to background, and asymmetry of ED (location of spikes in one direction from background signal); presence of small wave between spikes was additional but not obligatory sign. The SS were extracted the same way. Recognition was based on parameters: smooth onset and offset (spindle form), irregularity of spikes (including irregular interval between spikes and form of spikes), and symmetry (location of spikes in both directions from background signal). The second and the third experts performed a blind markup (without knowing the results of each other) of the EEG parts, which were randomly shuffled. As a result, the second expert extracted 233 EEG parts from 24 h record, and 127 parts were classified as discharges, while 106 were classified as spindles. The third expert marked 123 EEG parts as discharges and 35 as spindles. 75 parts could not be accurately classified. 96.5% matching ratio was achieved after comparison of the results. Only full-match cases were selected as train data for further analysis.

17.2.2 *Event detection algorithm*

The eight-order Butterworth filter bandpass filter [12] is used to select EEG 1.5–40 Hz range. In this chapter, the continuous Morlet wavelet transform was used [8]. The power spectrum density (PSD) of a time–frequency signal is calculated according to function (17.1):

$$S_x = |W(\tau, f)|^2 . \tag{17.1}$$

The continuous wavelet transform is defined by formula (17.2):

$$W(\tau, f) = \sqrt{f} \int x(t)\psi^*((t-\tau)f)dt. \tag{17.2}$$

In formula (17.2), x(t) refers to the source signal and $\Psi(\eta)$ refers to the Morlet mother function:

$$\psi(\eta) = \frac{1}{\sqrt{\pi F_b}} e^{2i\pi F_c \eta} e^{-\frac{\eta^2}{F_b}}. \tag{17.3}$$

It was proposed to use the above equation with $F_c = F_b = 1$.

Event Detection Algorithm (EDA) is based on the wavelet ridge power spectral density analysis [19, 20]. The ridge of wavelet spectrogram consists of points PSD(t_i, f_j)) with maximal power spectral

density at a time t_i:

$$R_i = \max(PSD(t_i, f)), \quad f \in [1.5, 40]. \tag{17.4}$$

Figure 17.1 shows the example of the ridge of EEG wavelet spectrogram.

Each ridge contains points related to the background activity. On the contrary, suspicious activities (further referred to as Events) can include SS and ED. First, points with sufficiently small values of PSD were removed. For this purpose, histograms of PSD values in the ridge of wavelet transform (Figure 17.2) were constructed.

Both events and background may contain points with small values of the ridge PSD. One can see in the histogram (Figure 17.2) that majority of ridge points are in the region of low PSD values. Some of these points do not refer to background activity. Therefore, the groups of nearest points in the ridge were considered. Number of events is shown as the function of threshold value $T(h_k) \in [\min(R);$

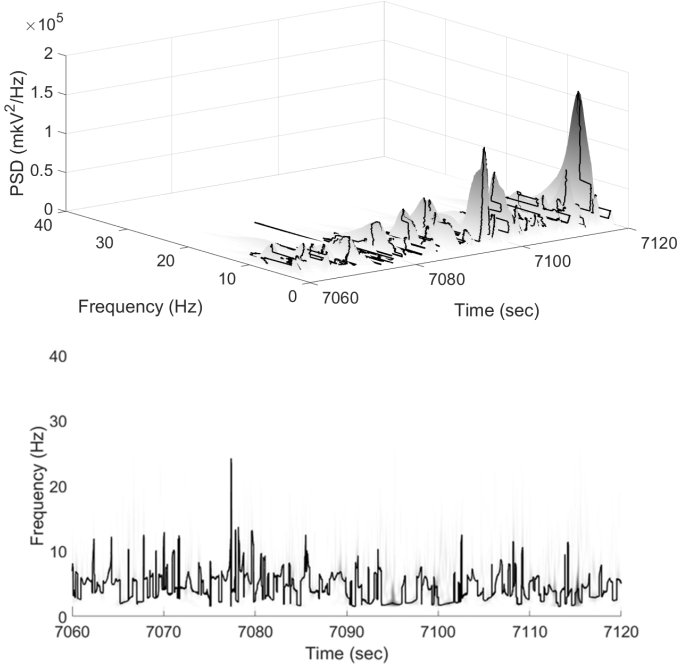


Figure 17.1. A ridge on a wavelet spectrogram (3D view and top view).

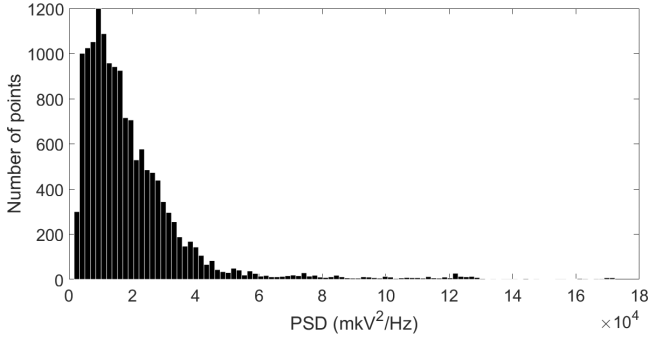


Figure 17.2. A PSD histogram of the points of the ridge.

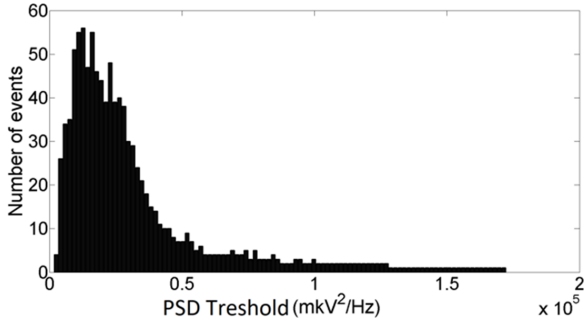


Figure 17.3. The histogram of the number of detected events.

$\max(R)]$ on Figure 17.3. For each value of the threshold $T(h_k)$, the points of the ridge that $(R_i > T(h_k))$ are selected. The maximum of this function is considered at the threshold value $T(h_{\max})$. The main goal of choosing such threshold is the identification of all events that can be attributed to suspicious activity, regardless of its type.

One of the event's features is duration. To find approximate duration, all events were prolonged until PSD values at the end and the beginning of the event reached the nearest local minima of the ridge. Afterward, events were selected in accordance with the rules provided by experts. In practice, a delay of less than 0.15 s is almost invisible to the eye. Therefore, events with a delay of less than 0.15 s were combined into one event. Thus, the number of events decreased, and their duration increased. However, events with a short duration were still present. In this chapter, events with a duration of less than 0.5 s

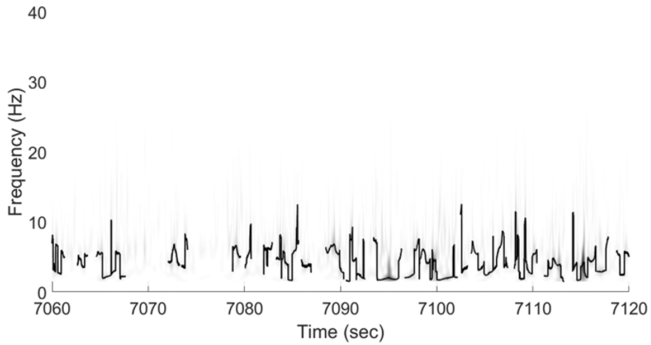


Figure 17.4. Selected by adaptive PSD threshold (see Figs. 17.2 and 17.3) time-frequency events on the wavelet spectrogram.

were not considered. The reason is that usually sleep spindles and epileptiform discharges last more than a half second. This correction was added to the algorithm (Figure 17.4). This is the last stage of the EDA.

The idea of EDA is to transition from raw EEG into time–frequency domain using the wavelets transform. The ridge points related to the suspicious activity were combined into events, and these events were selected in accordance with additional rules.

17.2.3 *Epilepsy prediction modeling and results*

The proposed event detection approach was applied to previously marked EEG records. As a result, a dataset of 365 observations was created, where 198 of the observations were labeled as ED and 167 as SS. We would refer to the binary variable as the target, which indicates a class of events according to markup: 1 in case of epileptiform activity and 0 in case of normal (SS) observation. Each detected event represents PSD and Frequency over time, while the duration of the event can be different among events. Table 17.1 shows the proposed features and their description. As discussed with the experts, these features can potentially indicate the presence of epileptiform activity. Based on this feature space, further analysis was done.

Obviously, it is proposed to use only important features in the classification model. Thus, feature importance analysis was done by plotting the Receiver Operating Characteristics (ROC) curve and

Table 17.1. List of EEG features and their AUC values.

Feature	Description	AUC
$f_{\min}$	Minimal Frequency of event ridge	0.52
$f_{\max}$	Maximal Frequency of event ridge	0.51
Δf	$f_{\max} - f_{\min}$	0.51
$P_{\min}$	Minimal PSD of event ridge	0.72
$P_{\max}$	Maximal PSD of event ridge	0.53
ΔP	$P_{\max} - P_{\min}$	0.50
mean(f)	Average Frequency of event ridge	0.61
std(f)	Frequency Standard Deviation of event ridge	0.64
std(f)/mean(f)	Standard Deviation to Mean ratio of Frequency ridge	0.67
$t(P_{\max})$	Relative Time of maximal PSD	0.65

measuring the Area Under Curve (AUC). From Table 17.1, one can also see the resulted AUC values for each proposed feature.

Features like $P_{\min}$, std(f)/mean(f), $t(P_{\max})$ are the most important, while $f_{\max}$, $f_{\min}$, and Δf have no correlation with epileptiform activity. Finally, only important features were used to train a classifier, so that the AUC of the feature is more than 0.6.

It was proposed to train a linear classifier due to (i) a low number of observations and (ii) uncertainty in the target variable, which is based on human diagnosis. The usage of nonlinear models would most likely lead to overfitting. That is why a logistic regression model was used. Logistic Regression model is a binary classification model, which estimates features weights (or coefficients) as in probability formulas (17.5), (17.6) [17]:

$$f(z) = \frac{1}{1 + e^{-z}}, \tag{17.5}$$

$$P\{y = 1 | \, x\} = f(z) = a_1 x_1 + \cdots + a_n x_n. \tag{17.6}$$

In the formula above, $f(z)$ is the probability of a positive outcome (in our case, epileptiform activity of event) and x_i refers to feature i value. The estimation of weights is done by maximizing the likelihood function using the gradient descent algorithm.

To validate the model, it was proposed to perform multiple train/test split methods. The dataset was randomly divided into train and test parts in 70/30 proportion. Coefficients evaluation was performed on a train subset, and prediction was made on the test

subset. As a result, AUC was measured on validated data. This training and testing procedure was done 1000 times to reach statistically stable results. After 1000 models were created, AUC for each model was estimated on a test subset. As a result, the distribution of AUC was analyzed. Table 17.2 shows AUC statistics: minimal and maximal values of distribution, mean and median, as well as 1st and 3rd quartiles of distribution. Median AUC is nearly 80%. Average coefficients of predictors are provided in Table 17.3. In this table, intercept is in addition to z in formula (17.6), which is not multiplied by any feature value. Figure 17.5 shows the distribution of AUC after simulations and Figure 17.6 shows the model ROC curve.

From Figure 17.6 and Table 17.2, one can see that the models' AUC lies in the range of 0.769–0.822, while there are some outliers with AUC less than 0.65 and more than 0.85. The cause of these outliers can be relatively small number of events in the train set. Thus, on average, classification accuracy of 80% can be expected from the proposed models.

17.3 Classification Models of Early-Stage Parkinson's Disease in Electroencephalography Features Space Methods

Morlet wavelet transform was used for time–frequency EEG features' analysis of the early-stage PD [18, 25]. The investigated groups consisted of 31 non-treated PD patients with the clinical

Table 17.2. Summary of models AUC values.

	Min.	1st Quartile	Median	Mean	3rd Quartile	Max.
AUC	0.643	0.769	0.796	0.795	0.822	0.896

Table 17.3. Average features coefficients in logistic regression models.

Intercept	$P_{\min}$	mean(f)	std(f)	std(f)/mean(f)	$t(P_{\max})$
−1.8203	−3.682e−05	0.5165	−1.5043	1.9391	0.6578

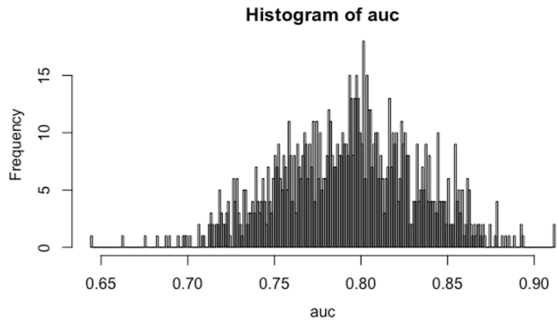


Figure 17.5. Histogram of AUC over simulations.

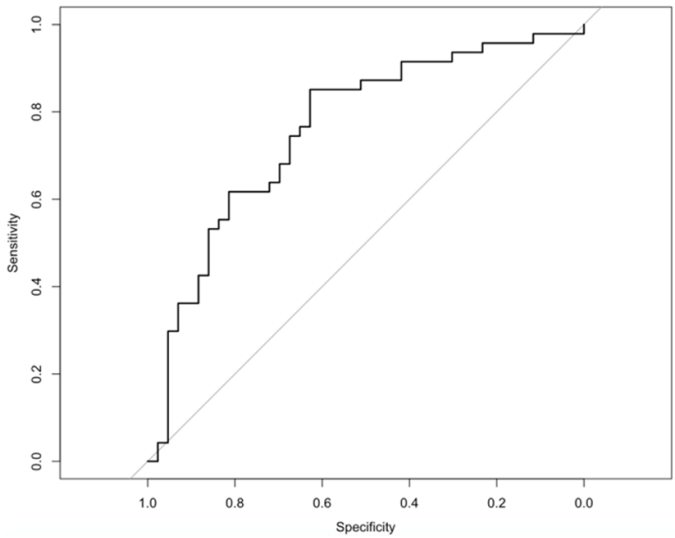
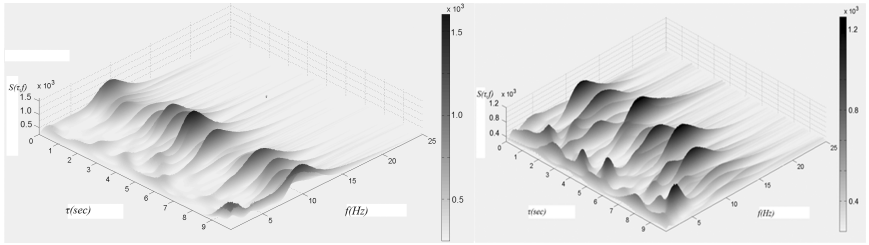


Figure 17.6. Sample model ROC curve, where sensitivity is true positive rate and specificity is false positive rate.

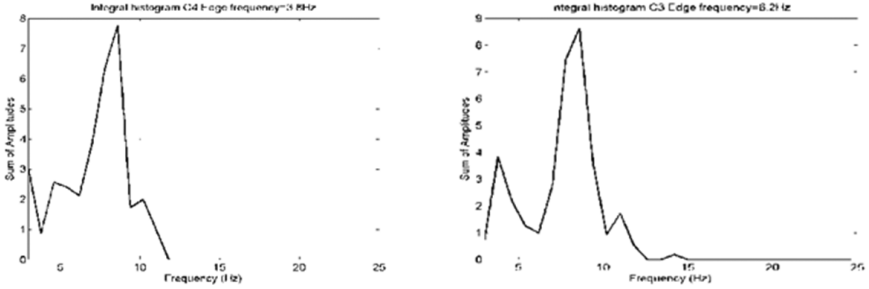
diagnosis of 1st stage according to the Hoehn–Yahr scale [10] and 18 control volunteers. Prediction results were tested on different sets of 22 PD patients and 16 normal volunteers. Special attention was paid to theta rhythm of EEG (4–6 Hz) and disorder of alpha rhythm (8–12 Hz).

Figure 17.7 illustrates the difference of $S(\tau)$ in the brain motor zone C3 (according to the standard 10×20 scheme of electrodes layout) for the normal volunteer (a) and (b) the patient at the first PD stage according to the qualitative stages of PD described by



(a) (b)

Figure 17.7. Time–frequency power density spectrogram of normal volunteer (a) and of the 1st stage PD patient (b) of the EEG signals in motor cortex zone C3.



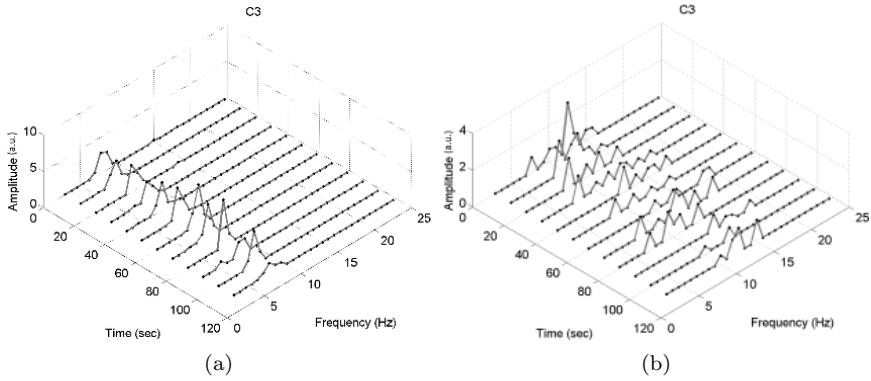
(a) (b)

Figure 17.8. Sum of extrema histograms in (0.7 Hz, 180 s) rectangles for symmetrical C3 (a), and C4 (b) EEG electrodes of 1st stage PD patient.

Hoen–Yahr. It can be seen that the patient's spectrogram shows high disorder, especially in the 8–12 Hz frequency range. Moreover, a theta rhythm arises in the low-frequency range. In order to analyze those features, it is proposed to consider the time–frequency distribution of spectrogram extrema.

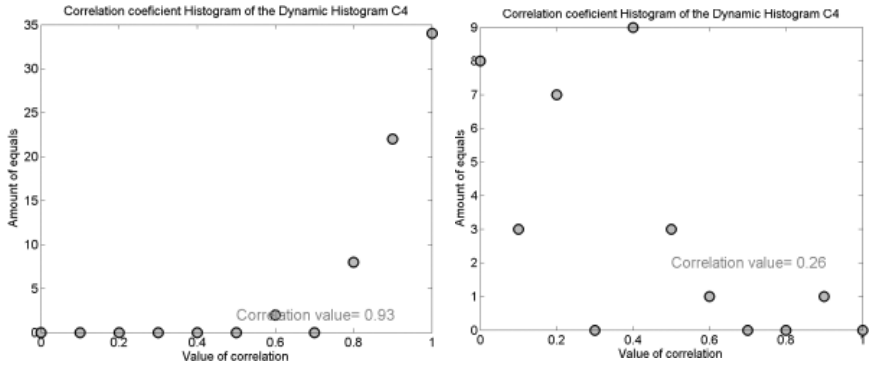
The distribution of extrema in time–frequency buckets (Δf, Δt) is used for further quantitative analysis. Figure 17.8 frequency shows histograms asymmetry in the 4–5 Hz frequency range of the 1st-stage PD patient. The existence of theta rhythm can be found in C3, while there is an absence of such rhythm in C4 electrode. This one-sided development of disease is common in the early stages.

The relation of theta rhythm amplitude to alpha rhythm amplitude can be considered as the feature of PD. Moreover, extrema histograms can be used for quantitative analysis of the dominant alpha rhythm disorder. The dynamical histograms calculated in (0.7 Hz,



(a) (b)

Figure 17.9. Dynamical histograms of the normal volunteer (a), and for the 1st stage PD patient. Histograms were calculated in (0.7 Hz, 10 s) rectangles.



(a) (b)

Figure 17.10. Histograms of correlation values for the normal volunteer (a) and the 2nd stage PD patient (b).

10 s) rectangles show the disorder of electrical activity for the 1st stage PD (Figure 17.9).

Such disorder can be evaluated with the correlation matrixes. Thus, the average correlation for normal volunteers would be higher than for PD patients. This evaluation can be done by histograms of correlation values. Figure 17.10. shows the difference between such histograms for the normal volunteer and the 2nd-stage PD patient.

Average value of correlation coefficients as well as its standard deviation can be considered as the features of PD.

Table 17.4. AUC values for logistic regression model in 8 pairs of electrodes.

	FP1FP2	F3F4	C3C4
PD hemisphere	0.82	0.79	0.73
Normal hemisphere	0.79	0.67	0.69

T3T4	F7F8	P3P4	T5T6	O1O2
0.69	0.79	0.60	0.68	0.56
0.63	0.60	0.62	0.62	0.68

Finally, EEG feature space P_i was created [25]:

$$\left| \begin{array}{l} P \in \{A_\theta/A_\alpha(j), A_\theta/A_\alpha(j^*), r(j), r(j^*), \sigma(j), \sigma(j^*)\} \\ \quad j, j^* \in \{Fp_1, Fp_2\}(F3, F4), (C3, C4), (T3, T4), \\ \qquad\qquad (P3, P4), (T5, T6), (O1, O2)\} \end{array} \right. . \quad (17.7)$$

Here A_θ/A_α (j) and A_θ/A_α (j^*) refer to the ratio of theta rhythm to alpha rhythm in two opposite hemispheres. Indexes j and j^* are 8 couple symmetrical EEG electrodes of standard 10×20 layout. By j we will refer to the hemisphere with PD and j^* as the normal hemisphere. The average and deviation of correlation coefficients are marked as $r(j)$, $r(j^*)$ and $\sigma(j)$, $\sigma(j^*)$, respectively.

To aggregate the features into one value, which could indicate the probability of PD, it was proposed to build a logistic regression model. The probability of outcome is linked to the linear prediction function:

$$\ln\left(\frac{p_i}{1-p_i}\right) = \beta_0 + \beta_1 x_{1,i} + \cdots + \beta_m x_{m,i}. \quad (17.8)$$

In this function, p_i is the probability of positive outcome for observation i, given $x_{m,i}$ — the feature m in a dataset. The weight of each feature can be computed by maximizing the likelihood function.

17.4 Clinical Results

Logistic regression model was trained on the EEG data of 31 patients with clinical diagnosis of 1st-stage non-treated PD and 18 control

volunteers. EEG signals were measured in 8 pairs of electrodes. The following table indicates the AUC results of the models for each pair of electrodes based on the feature space (1874) for PD hemisphere and normal hemisphere:

It can be seen that the best prediction accuracy is reached in Fp1 and Fp2 electrodes, while the worst in O1O2 and P3P4. Moreover, prediction accuracy on the PD hemisphere is slightly higher than on the normal hemisphere. Further, the model trained on the PD hemisphere was used. In total, 8 models were trained, and the outcome of the models was the probability of PD.

The model analysis was also done on the test data of 22 PD patients and 16 normal volunteers. EEG data of this group was not used in model training. Each model predicted probability of PD in every electrode. To compose all predictions in one value, it is necessary to develop an aggregation function based on each probability. It was proposed to use function F, which can be calculated according to the formula

$$F = \sum_{j,j^*} |P(j, j^*) - 0.5| \, \mathrm{AUC}(j, j^*). \tag{17.9}$$

This function summarizes all probabilities $P(j, j^*)$ are adjusted on 50%. This adjustment is needed, so the contribution is zero for not certain electrodes with a 50% probability of PD. Then these adjusted probabilities are weighted with AUC of a hemisphere symmetrical electrodes from Table 17.4. Thus, models with lower accuracy would not have the same weight as good models. In addition, electrodes O1O2 and P3P4 were excluded from this aggregation due to low AUC.

Figure 17.11 shows the distribution of F among patients with PD (stage = 1) and normal volunteers (stage = 0).

Patients with PD have higher values of F, which are mostly concentrated around 1. Although, normal volunteers have low values of F but with high deviation. To make a certain prediction, whether the person has PD or not, it is necessary to select a cut-off value. In accordance with this cut-off value, a person would be classified as PD or not. For sure, the cut-off value should have the optimal accuracy of classification among PD patients and among control volunteers. It is proposed to calculate recall of PD (equation (17.10)) and control

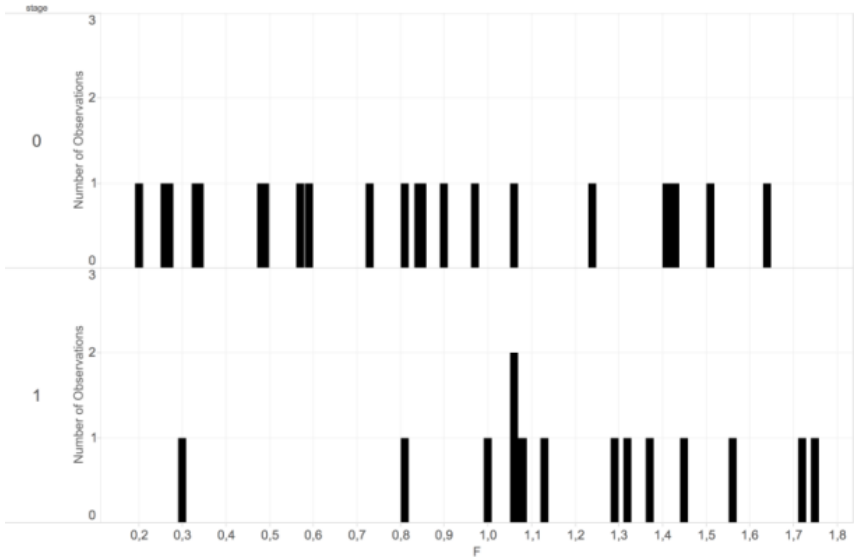


Figure 17.11. Distribution of aggregated function F for control (above) and PD patients (below).

(equation (17.11)), respectively:

$$\text{Re call}_1 = \frac{\text{TP}}{\text{TP} + \text{FN}}, \tag{17.10}$$

$$\text{Re call}_0 = \frac{\text{TN}}{\text{TN} + \text{FP}}. \tag{17.11}$$

In the formulas above, TP indicates the number of True Positives (correct prediction of PD), TN refers to True Negative (correct prediction of control), FP refers to False Positives (false prediction of PD), and FN refers to False Negatives (false prediction of control). For each possible cut-off value, these recall functions are calculated.

The results are shown in Figure 17.12. In case of low threshold, the model would assume everyone to have PD, thus recall of PD would be 100%, but recall of control is 0%. On the other hand, if the threshold is too high, then the model predicts everyone as control, but none as PD. The optimal threshold is around 1.1. In this case, both recalls would be near 73%. Overall accuracy is also present on the chart in circles.

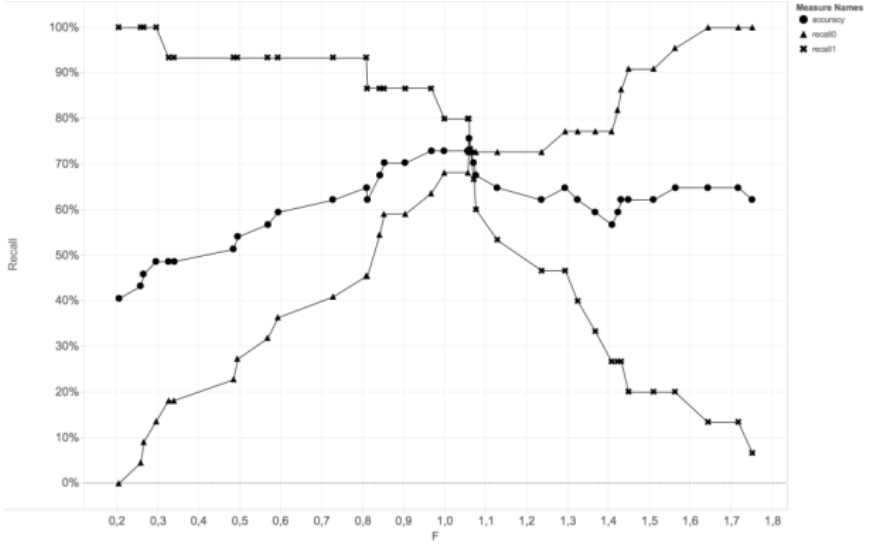


Figure 17.12. Recall curve of PD (cross) and control (triangle). Accuracy curve (circles) for various cut-off thresholds.

17.5 Conclusions

This chapter describes methods for EEG epileptiform activity recognition in laboratory rats before and after traumatic brain injury (TBI). Experts in neurobiology performed a markup of signals into ED and SS to make a dataset for training the model. A new method that allows finding suspicious activity in EEG signals was developed: Event Detection Algorithm. In this chapter, an algorithm that locates suspicious activity is described. The main advantages of the proposed method are automation and flexibility. The application of Morlet wavelet transform to raw EEG records allows obtaining information on detected fragments and identification of suspicious electrical activity. In addition to event detection, the algorithm estimates their time–frequency characteristics. This allows further classification of Events detected in the signal. PSD and Frequency features were extracted from EEG and formed a feature set for classification. On a resulting dataset, a logistic regression model was built with a binary target variable of epilepsy. However, only important factors were considered in the model. Models were validated on different

subsets of data, and performance was measured in AUC. The proposed algorithm of ED and SS recognition shows good performance results with an average of 80% accuracy of prediction. Further, this approach can be applied to the identification of epileptic activities in a long-term EEG record. For instance, long EEG records can be automatically processed, and parts with a high risk of epilepsy could be identified with the proposed model. This chapter shows the results of the model's application to the recognition of epileptiform activity in TBI rats and false surgery rats. The analysis shows that the distribution of ED probabilities after injury significantly shifts to the area of high probabilities, while the distribution does remain stable for false surgery rats. Moreover, majority of ED cases were found in some EEG channels, which corresponds to the common understanding of epileptiform activity development.

The EEG features of Parkinson's disease (PD) are analyzed in this chapter. It was shown that low-frequency theta rhythm of brain activity as well as disorder of dominant rhythm can be considered as the features of early-stage PD. The feature space was created, and the logistic regression model was built with the binary target of disease. The training data consisted of 31 patients with clinical diagnosis of 1st stage non-treated PD and 18 control volunteers. The model was trained for each of the 16 channels, and the accuracy AUC was measured in each channel respectively. The probabilities of disease among channels were summarized in the function in accordance with AUC. The resulting function was proven to have high accuracy of prediction on the test data of 22 PD patients and 16 normal volunteers. The accuracy of prediction was around 73%. The future scope of work covers the extraction of other potentially important features of PD from EEG data, testing other classification models, and aggregate functions to increase the accuracy of prediction.

The results of EEG signal analysis, as well as feature extraction techniques and model results, prove that the proposed approach can be applicable to Parkinson's disease diagnostics in the early stage, where the disease is already developed, but the clinical symptoms have not yet appeared.

Acknowledgments

The authors express their gratitude to prof. N.V. Gulyaeva, Dr. A.O. Tishkina dr., O.S. Sushkova, and Dr. A.V. Karabanov for presentation of the experimental data and discussions of the results.

References

[1] Aarabi, A., Wallois, F., and Grebe R. (2006) Automated neonatal seizure detection: A multistage classification system through feature selection based on relevance and redundancy analysis. *Journal of Clinical Neurophysiology*, 117(2), 328–340.

[2] Aarabi, A., Fazel-Rezai, R., and Aghakhani, Y. (2009) EEG seizure prediction: Measures and challenges. *Proceedings of the Annual International Conference of the IEEE Engineering in Medicine and Biology Society* (IEEE), pp. 1864–1867.

[3] Acharya, U.R., Hagiwara, Y., and Adeli, H. (2018) Automated seizure prediction. *Journal of Epilepsy & Behavior*, 88, 251–261.

[4] Alotaiby, T.N., Alshebeili, S.A., Alshawi, T., Ahmad, I., and El-Samie, F.EA. (2014) EEG seizure detection and prediction algorithms: A survey. *Journal on Advances in Signal Processing*, 183, 1–21.

[5] Berendse, H.W. and Stam, C.J (2007) Stage-dependent patterns of disturbed neural synchrony in Parkinson's disease. *Journal of Parkinsonism and Related Disorders*, 13(3), 440–445.

[6] Blanco, S., Kochen, S., Rosso, O.A., and Salgado, P. (1997) Applying time-frequency analysis to seizure EEG activity. *Journal of IEEE Engineering in Medicine and Biology Magazine*, 16(1), 64–71.

[7] Dingle, A.A., Jones, R.D., Carroll, G.J., and Fright, W.R. (1993) A multistage system to detect epileptiform activity in the EEG. *Journal of IEEE Transactions on Biomedical Engineering*, 40(12), 1260–1268.

[8] Goupillaud, P., Grossmann, A., and Morlet, J. (1984) Cycle-octave and related transforms in seismic signal analysis. *Journal of Geoexploration*, 23(1), 85–102.

[9] Greene, B.R., Faul, S., Marnane, W.P., Lightbody, G., Korotchikova, I., and Boylan, G.B. (2008) A comparison of quantitative EEG features for neonatal seizure detection. *Journal of Clinical Neurophysiology*, 119(6), 1248–1261.

[10] Hoehn, M.M. and Yahr, M.D. (1967) Parkinsonism: Onset, progression and mortality. *Journal of Neurology*, 17, 427–442.

[11] Hramov, A.E., Koronovskii, A.A., Makarov, V.A., Pavlov, A.N., and Sitnikova, E. *Wavelets in Neuroscience* (Springer, Berlin).

[12] Ifeachor, E.C and Jervis, B.W. (2002) *Digital Signal Processing: A Practical Approach* (Pearson Education, Prentice Hall, Harlow, England).

[13] Kabadi, S.V., Hilton, G.D., Stoica, B.A., Zapple, D.N., and Faden, A.I. (2010) Fluid-percussion-induced traumatic brain injury model in rats. *Journal of Nature Protocols*, 5(9), 1552–1563.

[14] Kannathal, N., Choo, M.L., Acharya, U.R., and Sadasivan, P.K. (2005) Entropies for detection of epilepsy in EEG. *Journal of Computer Methods and Programs in Biomedicine*, 80(3), 187–194.

[15] Libenson, M.H. (2009) *Practical Approach to Electroencephalography* (Elsevier Health Sciences, Saunders Elsevier, USA, Philadelphia, PA).

[16] Mallat, S. (2009) *A Wavelet Tour of Signal Processing* (Elsevier Inc., Elsevier, USA, San Diego, California).

[17] Menard, Scott W. (2002) *Applied Logistic Regression Analysis*, 2nd. Ed. (Sage Publications Inc., USA, Thousand Oaks, California).

[18] Obukhov, Y.V., Maliuta, I.A., and Obukhov, K.Y. (2016) Classification of early stage Parkinson's disease in electroencephalography features space. *Journal of Pattern Recognition and Img. Analysis*, 26(4), 810–816.

[19] Obukhov, K., Kershner, I., Komoltsev, I., and Obukhov, Y. (2018) Machine learning applications to epileptiform activity recognition in rats after traumatic brain injury. *Proceedings of 31st IEEE International Symposium on Computer-Based Medical Systems*, CBMS, pp. 59–64.

[20] Obukhov, K., Kersher, I., Komoltsev, I., and Obukhov, Y. (2018) Epileptiform activity detection and classification algorithms of rats with post-traumatic epilepsy. *Journal of Pattern Recognition and Image Analysis*, 28(2), 346–353.

[21] Paivinen, N., Lammi, S., Pitkanen, A., Nissinen, J., Penttonen, M., and Gronfors. T. (2005) Epileptic seizure detection: A nonlinear viewpoint. *Journal Computer Methods and Programs in Biomedicine*, 79(2), 151–159.

[22] Saab, M. and Gotman, J. (2005) A system to detect the onset of epileptic seizures in scalp EEG. *Journal of Clinical Neurophysiology*, 116(2), 427–442.

[23] Stoffers, D., Bosboom, J.L.W., Deijen, J.B., Wolters, E.C., Berendse, H.W., and Stam, C.J. (2007) Slowing of oscillatory brain activity is a stable characteristic of Parkinson's disease without dementia. *Journal of Brain*, 130(7), 1–14.

[24] Subasi, A. (2006) Automatic detection of epileptic seizure using dynamic fuzzy neural networks. *Journal of Expert Systems with Applications*, 31(2), 320–328.

[25] Sushkova, O.S., Gabova, A.V., Karabanov, A.V., Kershner, I.A., Obukhov, K.Y., and Obukhov, Y.V. (2015) Time-frequency analysis of simultaneous measurements of electroencephalograms, electromyograms, and mechanical tremor under Parkinson disease. *Journal of Communications Technology and Electronics*, 60(10), 1109–1116.

[26] Ugrumov, M.V., Khaindrava, V.G., Kozina, E.A., Kucheryanu, V.G., Bocharov, E.V., Kryzhanovsky, G.N., Kudrin, V.S., Narkevich, V.B., Klodt, P.M., and Raevsky, K.S. (2011) Modeling of preclinical and clinical stages of Parkinson's disease in mice. *Journal of Neuroscience*, 181, 175–188.

[27] van Vugt, M.K., Sederberg, P.B., and Kahana, M.J. (2007) Comparison of spectral analysis methods for characterizing brain oscillations. *Journal of Neuroscience Methods*, 162(1–2), 49–63.

[28] Van Zaen, J., Murray, M.M., Meuli, R.A., and Vesin, J.-M. (2013) Adaptive filtering methods for identifying cross-frequency couplings in human EEG. *Journal of PLOS One*, 8(4), e60513.

[29] Warby, S.C., Wendt, S.L., Welinder, P., Munk, E.G.S., Carrillo, O., Sorensen, H. B.D., Jennum, P., Peppard, P.E., Perona, P., and Mignot, E. (2014) Sleep-spindle detection: Crowdsourcing and evaluating performance of experts, non-experts and automated methods. *Journal of Nature Methods*, 11(4), 385–392.

© 2025 World Scientific Publishing Company
https://doi.org/10.1142/9789811267215_0018

Chapter 18

Search of Completely Verified Regularities in Biomedical Data

Oleg V. Senko[*,‡] **and Anna V. Kuznetsova**[†,§]

[*]*The Federal Research Center "Computer Science and Control" of the Russian Academy of Sciences, Moscow, the Russian Federation*
[†]*The Institute for Biochemical Physics of Russian Academy of Sciences, Moscow, the Russian Federation*
[‡]*senkoov@mail.ru*
[§]*azfor@yandex.ru*

A method is discussed that is aimed to search statistically valid regularities in data. Each regularity is the partition of explanatory variable space providing optimal separation of cases with different levels of target variable Y. Each regularity is verified with the help of permutation tests including searching of optimal partitions at random datasets where target Y values are randomly rearranged relatively fixed positions of explanatory variables. The concept of complete verification aimed to meet the redundancy problem is used when more complicated regularities are tested. Special optimal partitioning techniques were developed to assess the difference between regularities in two datasets or to find cyclic variations of Y in time series. Examples of several regularities from different applied tasks are represented.

Keywords: Optimal partitioning, permutation test, completely verified regularities, multiple testing, cyclic variations.

18.1 Introduction

Assessment of explanatory variables' effects on the outcome is one of the most important tasks in many research. Various pattern recognition or regression methods may be used for this purpose. Let us note that the main goal of classification or regression techniques is the best prediction of Y by explanatory variables. Best forecasting ability may be achieved for some set of selected informative regressors. At that other variables are ignored in the corresponding solving rule. But for many purposes, it is interesting to describe possibly all existing data statistically valid effects of explanatory variables on Y. Such a task may be partly solved with the help of statistical tests or ANOVA. However, the goal of statistical tests is the evaluation of the validity of existing correlations or differences between groups of observations. So additional tools are needed that would allow us to recover and describe efficiently statistically valid dependencies. One of the possible approaches is searching such subregions of explanatory variables space where levels of dependent variable Y decline significantly from Y mean in whole dataset or at least in neighbor subregions. Tasks of this type may be solved with the help of classification or regression trees [2] including classification trees that implement two-dimensional partitioning [10] or with the help of logical regularities techniques [18]. Let us note that trees methods usually implement splits that in the best way improve capability of recognition or forecasting. At that, some alternative splits are omitted. So search of regularities is not exhaustive. Besides classification or regression trees are not focused on validation of found regularities. In this chapter, optimal valid partitioning (OVP) approach [13, 14] is discussed that is aimed at revealing regularities in datasets that are associated with the effect of variables on outcome variable Y. The goal of OVP analysis is to receive existing data as a possible complete and valid description of the relationship between Y and potential explanatory variables $X_1, \ldots, X_n$. This approach is based on searching partitions of explanatory variables space $\mathbf{R}_x$ with best possible separation of objects with different levels of outcome variable Y. So elements of optimal partitions are regions in explanatory variables space where target variable Y levels deviate significantly from Y mean in whole training set or at least in neighboring regions.

18.2 Optimal Partitioning

Outcome variable Y may be a binary indicator of two classes of objects or it may be a continuous scalar variable. Also Y may be binary or continuous vector or survival function. Optimal partitions are searched inside fixed partition family by empirical (training) dataset $\tilde{S}_t = \{s_1 = (y_1, \mathbf{x}_1), \ldots, s_m = (y_m, \mathbf{x}_m)\}$, where y_j is part of object s_j description related to Y and $\mathbf{x}_j$ is vector of explanatory variables for dataset. In task when Y is a scalar or vector variable, y_j is simply a scalar or vector value of Y on object s_j. In survival tasks, y_j is pair (t_j, α_j), where t_j is the last observation time and α_j is the binary indicating if object really exists at time point t_j.

Partition families of the first type are defined as sets of partitions with a limited number of elements that are constructed by the same procedure. The one-dimensional and two-dimensional families of the first type are considered. The one-dimensional families include partitions of admissible intervals of single variables. The simplest Family I (Figure 18.1) includes all partitions with two elements that are divided by one boundary point. The more complex Family II (Figure 18.1) includes all partitions with no more than three elements that are divided by two boundary points. The two-dimensional Family III includes all partitions of two-dimensional admissible areas

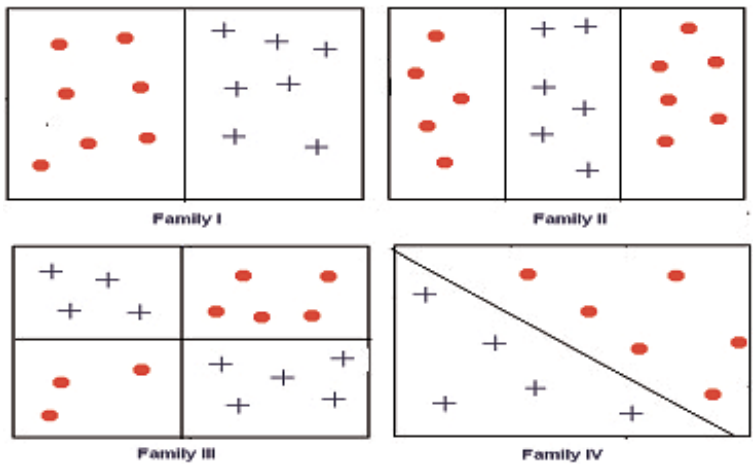


Figure 18.1. One-dimensional and two-dimensional partition families used in OVP method.

with no more than four elements that are separated by two boundary lines parallel to coordinate axes. The two-dimensional Family IV includes all partitions of two-dimensional admissible areas with no more than two elements that are separated by one boundary straight line. The examples of partitions from each of the three abovementioned families are given at Figure 18.1. It may be noted that the families of partitions I–IV are also used in the method of pattern recognition "Multi-Model Statistically Weighted Syndromes" [8, 15].

The families of the second type include partitions that are constructed by previously found partitions of $\tilde{S}_t$. Partition $\{q_1, \ldots, q_r\}$ of $\mathbf{R}_x$ is calculated by partition $\{\tilde{s}_1, \ldots, \tilde{s}_r\}$ of $\tilde{S}_t$ with the help of following simple rule: point $\mathbf{x}' \in \mathbf{R}_x$ is put to element q_i if minimal distance between $\mathbf{x}'$ and descriptions of objects from $\tilde{s}_i$ is less than corresponding distances for subsets from $\{\tilde{s}_1, \ldots, \tilde{s}_l\} \backslash \tilde{s}_i$.

Quality functionals: Let $R = \{\tilde{s}_1, \ldots, \tilde{s}_r\}$ be the partition of dataset $\tilde{S}_t$. Integral quality functional at partition R is defined as

$$F_{\text{Int}}(\tilde{S}_t, R) = \sum_{i=1}^{r} \rho[\hat{y}(\tilde{s}_i), \hat{y}(\tilde{S}_t)] m_i,$$

where m_i is the number of objects in $\tilde{s}_i$ and $\hat{y}(\tilde{s}_i)$ is the mean value of Y inside $\tilde{s}_i$. In survival tasks, $\hat{y}(\tilde{s}_i)$ is the Kaplan–Mayer estimate of the survival curve calculated by $\tilde{s}_i$:

$$P(T, \tilde{s}_i) = \Pr_{K-M}(t \le T, \tilde{s}_i) = \prod_{\mathbf{x}_j \in \tilde{s}_i, t_j \le T} \left[\frac{m-j}{m-j+1}\right]^{\alpha_j}.$$

In case when Y is scalar variable, $\rho[\hat{y}(\tilde{s}_i), \hat{y}(\tilde{S}_t)] = [\hat{y}(\tilde{s}_i) - \hat{y}(\tilde{S}_t)]^2$. In case when Y is vector variable of dimension k,

$$\rho[\hat{y}(\tilde{s}_i), \hat{y}(\tilde{S}_t)] = \sum_{j=1}^{k} [\hat{y}_j(\tilde{s}_i) - \hat{y}_j(\tilde{S}_t)]^2.$$

In survival tasks,

$$\rho[\hat{y}(\tilde{s}_i), \hat{y}(\tilde{S}_t)] = \int_0^{T_i} [P(t, \tilde{s}_i) - P(t, \tilde{S}_t)]^2 dt,$$

where T_i is the time of last observation in subset $\tilde{s}_i$. Integration over interval $[0, T_i]$ is implemented because T_i is always not greater than

the time of the last observation in $\tilde{S}_t$. But at $[0, T_i]$, the survival curve $P(t, \tilde{s}_i)$ is better evaluated from the data.

Besides integral functional $F_{\text{int}}(\tilde{S}_t, R)$, local functional $F_{\text{loc}}(\tilde{S}_t, R)$ is possible that is defined as

$$F_{\text{loc}}(\tilde{S}_t, R) = \max_{i=1,\ldots,r} \rho[\hat{y}(\tilde{s}_i), \hat{y}(\tilde{S}_t)]m_i.$$

Unlike integral functional, local functional allows us to pick out the most distant from the remaining part of $\tilde{S}_t$ subregion of partition. Optimal values of quality functional in dataset will be further referred to as F_I^o or as F_L^o.

Optimal partitioning technique that is represented in this section may be used in tasks where Y is the scalar variable. It is based on use for evaluating a new partition results of evaluating of previous one. So it is more economical than direct recalculating.

18.2.1 *Search of optimal one-dimensional partitions*

Suppose that initial information is a set of pairs $\tilde{S}_t = \{(y_1, x_1), \ldots, (y_m, x_m)\}$, where $x_1, \ldots, x_m$ are values of scalar explanatory variable X. Assume that explanatory $X \in \{a_1, \ldots, a_r | a_i \in R, i = 1, \ldots, r\}$, where $a_1 < a_2 \cdots < a_r$ Boundary points are calculated as $b_1 = \frac{a_1+a_2}{2}, \ldots, b_{r-1} = \frac{a_{r-1}+a_r}{2}$. Then we calculate sets $\{\hat{y}_1, \ldots, \hat{y}_r\}$ and $\{m_1, \ldots, m_r\}$ corresponding to elements from $\{a_1, \ldots, a_r\}$, where m_i is the number of objects in $\tilde{S}_t$ with $X = a_i$, $\hat{y}_i = \sum_{j=1}^{m} y_j I(x_j, a_i)$, where $I(x_j, a_i) = 1$ if $x_j = a_i$ and $I(x_j, a_i) = 0$ otherwise. Suppose that optimal partition is searched inside simplest family 1. Optimal partitions searching inside Family I:

(a) At the initial step boundary quality functional is calculate for boundary point b_1. It forms the following subsets of $\tilde{S}_t$: left $\tilde{s}_l$ with $X < b_1$, right $\tilde{s}_r$ with $X > b_1$. Let $\hat{m}_l$ be the number of observations in $\tilde{s}_l$, $\hat{m}_r$ be the number of observations in $\tilde{s}_r$.
Calculate $\tilde{y}(\tilde{s}_l) = \frac{\sum_{a_i<b_1}^{r} \hat{y}_i}{\hat{m}_l}$, $\tilde{y}(\tilde{s}_r) = \frac{\sum_{a_i>b_2}^{\hat{y}_i}}{\hat{m}_r}$.
Then one of the quality functionals is calculated:

$$F_{\text{int}}(\tilde{S}_t, R) = \rho[\tilde{y}(\tilde{s}_l)\tilde{y}(\tilde{S}_t)]m_l + \rho[\tilde{y}(\tilde{s}_m)\tilde{y}(\tilde{S}_t)]m_m + \rho[\tilde{y}(\tilde{s}_r)\tilde{y}(\tilde{S}_t)]m_r$$

or

$$F_{\mathrm{Ioc}}(\tilde{S}_t, R) = \max\{\, \rho[\tilde{y}(\tilde{s}_l)\tilde{y}(\tilde{S}_t)]m_l, \rho[\tilde{y}(\tilde{s}_r)\tilde{y}(\tilde{S}_t)]m_r\}.$$

(b) Assume that values $\hat{m}_l$, $\hat{m}_r$, $\tilde{y}(\tilde{s}_l)$, $\tilde{y}(\tilde{s}_r)$ were used when quality functional for boundary points b_{i-1} was calculated. To receive quality functional for boundary points b_i, it is sufficient to take

$$\begin{aligned}
\tilde{y}(\tilde{s}_l) \;&\text{ equal }\; \frac{\tilde{y}(\tilde{s}_l)m_l + \hat{y}_i}{\hat{m}_l + m_i},\\
\tilde{y}(\tilde{s}_r) \;&\text{ equal }\; \frac{\tilde{y}(\tilde{s}_r)m_r - \hat{y}_i}{\hat{m}_l - m_i},\\
\hat{m}_l \;&\text{ equal }\; \hat{m}_l + m_i,\\
\hat{m}_r \;&\text{ equal }\; \hat{m}_r - m_i
\end{aligned}$$

and to recalculate quality functional with these new values.

(c) Procedure is repeated until all boundary points are tested.

18.2.2 *Optimal partitions searching inside Family II*

(a) At the initial step boundary, quality functional is calculated for a pair of boundary points b_1 and b_2. It forms three subsets of $\tilde{S}_t$: left $\tilde{s}_l$ with $X < b_1$, middle $\tilde{s}_r$ with $b_1 < X < b_2$, right $\tilde{s}_r$ with $X > b_1$.

As in the previous section, $\hat{m}_l$ is the number of observations in $\tilde{s}_l$, $\hat{m}_m$ is the number of observations in $\tilde{s}_m$, and $\hat{m}_r$ is the number of observations in $\tilde{s}_r$.

(b) Assume that values $\hat{m}_l$, $\hat{m}_m$, $\hat{m}_r$, $\tilde{y}(\tilde{s}_l)$, $\tilde{y}(\tilde{s}_m)$, $\tilde{y}(\tilde{s}_r)$ were used when quality functional for boundary points b_{i-1} and b_{j-1} were calculated. To receive quality functional for boundary points b_i and b_j, it is sufficient to take

$$\begin{aligned}
\tilde{y}(\tilde{s}_l) \;&\text{ equal }\; \frac{\tilde{y}(\tilde{s}_l)m_l + \hat{y}_i}{\hat{m}_l + m_i},\\
\tilde{y}(\tilde{s}_m) \;&\text{ equal }\; \frac{\tilde{y}(\tilde{s}_r)m_r - \hat{y}_i + \hat{y}_j}{\hat{m}_l - m_i + m_j},\\
\tilde{y}(\tilde{s}_r) \;&\text{ equal }\; \frac{\tilde{y}(\tilde{s}_r)m_r - \hat{y}_i - \hat{y}_j}{\hat{m}_l - m_i - m_j},
\end{aligned}$$

$$\hat{m}_l \text{ equal } \hat{m}_l + m_i,$$
$$\hat{m}_m \text{ equal } \hat{m}_m - m_i + m_j,$$

$\hat{m}_{lr}$ equal $\hat{m}_r - m_i - m_j$ and to recalculate quality functional with these new values.

18.2.3 *Two-dimensional partitions from Family III*

Suppose that initial information is a set of pairs $\tilde{S}_t = \{(y_1, x_{11}, x_{12}), \ldots, (y_m, x_{m1}, x_{m2})\}$, where $x_{1i}, \ldots, x_{mi}$ are values of scalar explanatory variable X_i, $i = 1, 2$. Assume that explanatory

$$X_1 \in \{a_1^1, \ldots, a_{r_1}^1 | a_i^1 \in R, i = 1, \ldots, r_1\} \quad \text{and}$$
$$X_2 \in \{a_1^2, \ldots, a_{r_2}^2 | a_i^2 \in R, i = 1, \ldots, r_2\},$$

where $a_1^1 < a_2^1 \cdots < a_{r_1}^1$ and $a_1^2 < a_2^2 \cdots < a_{r_2}^2$. At initial stage, boundary points are calculated as

$$b_1^1 = \frac{a_1^1 + a_2^1}{2}, \ldots, b_{r_1-1}^1 = \frac{a_{r_1-1}^1 + a_{r_1}^1}{2},$$
$$b_1^2 = \frac{a_1^2 + a_2^2}{2}, \ldots, b_{r_2-1}^2 = \frac{a_{r_2-1}^2 + a_{r_2}^2}{2}.$$

Then we calculate sets $\{\hat{y}_{i'i''} | \, i' = 1, \ldots, r_1, \, i'' = 1, \ldots, r_2\}$ and $\{m_{i'i''} | i' = 1, \ldots, r_1, \, i'' = 1, \ldots, r_2\}$ corresponding to pairs from $\{(a_{i'}^1, a_{i''}^2) | i' = 1, \ldots, r_1, \, i'' = 1, \ldots, r_2\}$, where $m_{i'i''}$ is number of objects in $\tilde{S}_t$ with $X_1 = a_{i'}^1$, $X_2 = a_{i''}^2$

$$\hat{y}_{i'i''} = \sum_{j=1}^{m} y_j I(x_{j1}, a_{i'}^1) I(x_{j2}, a_{i''}^2).$$

In the first step, quality functional is calculated for partition with boundary points b_1^1 for X_1 and b_1^2 for X_2:

$$F_{\text{int}}(\tilde{S}_t, R) = \rho[\tilde{y}(\tilde{s}_{ul})\tilde{y}(\tilde{S}_t)]m_{ul} + \rho[\tilde{y}(\tilde{s}_{ur})\tilde{y}(\tilde{S}_t)]m_{ur}$$
$$+ \rho[\tilde{y}(\tilde{s}_{ll})\tilde{y}(\tilde{S}_t)]m_{ll} + \rho[\tilde{y}(\tilde{s}_{lr})\tilde{y}(\tilde{S}_t)]m_{lr}$$

or

$$F_{\mathrm{loc}}(\tilde{S}_t, R) = \mathbf{max}\{\rho[\tilde{y}(\tilde{s}_{ul})\tilde{y}(\tilde{S}_t)]m_{ul}, \rho[\tilde{y}(\tilde{s}_{ur})\tilde{y}(\tilde{S}_t)]m_{ur},$$
$$\times \rho[\tilde{y}(\tilde{s}_{ll})\tilde{y}(\tilde{S}_t)]m_{ll}, \rho[\tilde{y}(\tilde{s}_{lr}), \tilde{y}(\tilde{S}_t)]m_{lr}\}.$$

Here following notations are used:

$\tilde{s}_{ul}$ is the upper left subset with $X_1 < b_1^1$ and $X_2 > b_1^2$,
$\tilde{s}_{ur}$ is the upper right subset with $X_1 > b_1^1$ and $X_2 > b_1^2$,
$\tilde{s}_{ll}$ is the lower left subset with $X_1 < b_1^1$ and $X_2 < b_1^2$,
$\tilde{s}_{lr}$ is the lower right subset with $X_1 > b_1^1$ and $X_2 < b_1^2$,
$\hat{y}_{**}$ is the sum of y-components in $\tilde{s}_{**}$; m_{**} is the number of observations in $\tilde{s}_{**}$

(a) Calculate

$$\tilde{y}(\tilde{s}_{ul}) = \frac{\tilde{y}^s_{ul}}{\hat{m}^s_{ul}},$$

where

$$\tilde{y}^s_{ul} = \sum_{i'=2}^{r_2} \bar{y}^l_{i'}, \quad \hat{m}^s_{ul} = \sum_{i'=2}^{r_2} \bar{m}^l_{i'}, \quad \tilde{y}(\tilde{s}_{ur}) = \frac{\tilde{y}^s_{ur}}{\hat{m}^s_{ur}},$$

where

$$\tilde{y}^s_{ur} = \sum_{i'=2}^{r_2} \bar{y}^r_{i'}, \quad \hat{m}^s_{ur} = \sum_{i'=2}^{r_2} \bar{m}^r_{i'}.$$

At that

$$\bar{y}^l_{i'} = y_{1i'}, \quad \bar{m}^l_{i'} = m_{1i'}, \quad \bar{y}^r_{i'} = \sum_{k=2}^{r_1} y_{ki'}, \quad \bar{m}^r_{i'} = \sum_{k=2}^{r_1} m_{ki'}.$$

(b) Assume that values $\hat{m}^s_{**}$, $\tilde{y}^s_{**}$ were used when quality functional for boundary points $b^1_{i'}$ and $b^2_{i''-1}$ were calculated. To receive quality functional for points $b^1_{i'}$ and $b^2_{i''}$ it sufficiently to take

$$\tilde{y}^s_{ll} \text{ equal } \tilde{y}^s_{ll} + \bar{y}^l_{i''}, \quad \hat{m}^s_{ll} \text{ equal } \hat{m}^s_{ll} + m^l_{i''},$$
$$\tilde{y}^s_{ul} \text{ equal } \tilde{y}^s_{ul} - \bar{y}^r_{i''}, \quad \hat{m}^s_{ul} \text{ equal } \hat{m}^s_{ul} - m^l_{i''}$$
$$\tilde{y}^s_{lr} \text{ equal } \tilde{y}^s_{lr} + \bar{y}^r_{i''}, \quad \hat{m}^s_{lr} \text{ equal } \hat{m}^s_{lr} + m^r_{i''},$$
$$\tilde{y}^s_{ur} \text{ equal } \tilde{y}^s_{ur} - \bar{y}^r_{i''}, \quad \hat{m}^s_{ur} \text{ equal } \hat{m}^s_{ur} - m^r_{i''}$$

and to recalculate all $\tilde{y}(\tilde{s}_{**})$ and quality functionals with these new values.

(c) Assume that values $\hat{m}^s_{**}$, $\tilde{y}^s_{**}$, $\bar{y}^l_*$, $\bar{y}^r_*$, $\bar{m}^l_*$, $\bar{m}^r_*$, were used when quality functional for boundary points $b^1_{i'-1}$ and $b^2_{i''}$ were calculated. To receive quality functional for points $b^1_{i'}$ and it is $b^2_{i''}$ sufficiently to take $\bar{y}^l_*$ equal $\bar{y}^l_* + \hat{y}_{i'*}$, $\bar{m}^l_*$ equal $\bar{m}^l_* + m_{i'*}$, $\bar{y}^r_*$ equal $\bar{y}^r_* - \hat{y}_{i'*}$, $\bar{m}^r_*$ equal $\bar{m}^r_* - m_{i'*}$, $\hat{m}^s_{lr} + m^r_{i''}$, $\tilde{y}^s_{ur}$ equal $\tilde{y}^s_{ur} - \bar{y}^r_{i''}$, $\hat{m}^s_{ur}$ equal $\hat{m}^s_{ur} - m^r_{i''}$. and to recalculate all $\tilde{y}(\tilde{s}_{**})$ q. Then values $\hat{m}^s_{**}$, $\tilde{y}^s_{**}$, and quality functionals with these new values and recalculated.

Procedure is continued until all possible boundaries are enumerated.

18.2.4 *Regularities verification by permutations based procedures*

The optimal partition is considered to be a statistically valid pattern after carrying out a procedure of verification. In the existing version of the OVP method, permutation tests (PTs) are used for verification. This type of statistical technique is becoming an increasingly popular way to estimate statistical validity as computing performance grows [1,5,6,11,12]. Permutation tests unlike their parametric counterparts allow to avoid assumptions about baseline distributions which are difficult to justify. Permutation tests are also not so sensitive to dataset sizes as distribution-free tests based on asymptotical convergence [12].

The basic version of the permutation test used to verify patterns from Family I will be referred to as PT1. PT1 is based on testing null hypothesis that target Y is independent of explanatory variable X. It is supposed that all objects in dataset are independent and belong to the same distribution.

Suppose that integral functional is used to evaluate partition quality. Algorithm for calculating p-values of optimal one-dimensional partition on a dataset $\tilde{S}_t = \{(y_1, x_1), \ldots, (y_m, x_m)\}$ includes the following steps:

1. Find on $\tilde{S}_t$ optimal partition of interval (a_l, a_r) with boundary point b^o_t. Value $F_{\text{int}}(\tilde{S}_t, b^o_t)$ is used as observed value of test statistics.
2. Take set $\tilde{\mathbf{S}}_{ir} = \{\tilde{S}^j_{\text{tr}} | j = 1, \ldots, \text{N}\}$ of random permutations of $\tilde{S}_t$. This is obtained by considering N independent random

permutations $\{\mathbf{f}_j = (f_1^j, \ldots, f_m^j) | j = 1, \ldots, \mathrm{N}\}$ of object labels $(1, \ldots, m)$. So $\tilde{\mathbf{S}}_{\mathrm{tr}} = \{\tilde{S}_{\mathrm{tr}}^j = \{(y_{f_1^j}, x_1), \ldots, (y_{f_m^j}, x_m) | j = 1, \ldots, \mathrm{N}\}$.

3. For each $j \in \{1, \ldots, \mathrm{N}\}$, find on $\tilde{S}_{ir}^j$ optimal partition of interval (a_l, a_r) with boundary point b_{jr}^o and calculate $F_{\mathrm{int}}(S_{\mathrm{tr}}^j, b_{jr}^o)$.
4. The estimate of p-value $\hat{p}$ is calculated as the proportion of datasets in $\tilde{\mathbf{S}}_{\mathrm{tr}}$ where the quality of partitions is better than in the true dataset:

$$\hat{p} = \frac{\#\{\tilde{S}_{ir}^j | F_{\mathrm{int}}(\tilde{S}_{\mathrm{tr}}^j, b_{jr}^o) \geq F_{\mathrm{int}}(\tilde{S}_t, b_t^o),\ j = 1, \ldots, \mathbf{N}\}}{\mathbf{N}}. \quad (18.1)$$

It was shown in [14] that estimate $\hat{p}$ is equal exact probability when all possible distinct permutations are enumerated. We used from 1000 to 3000 permutations to evaluate $\hat{p}$ by formula (18.1).

18.2.5 *Examples of one-dimensional regularity*

Figure 18.2 illustrates the dependence of radiation situation at International Space Station (ISS) on apogee altitudes [17]. The sign "o" marks the days when the level of 17.3 mrad was exceeded and the sign "+" marks the days with the level below 12.6 mrad. For the apogee >391.3, days with the daily increment of dose >17.3 prevail (360 days with increment >17.3 and 134 days with increment <12.6). However, for the apogee <391.3, the fraction of days with such levels of increment is rather small (86 days with the increment >17.3 and 228 days with the increment <12.6).

We can see from Figure 18.2 that boundary point 391.3 is crucial for daily dose rate. There were no dataset among two thousand randomly generated from initial dataset where F_{int} was greater than F_{int} at training set. So statistical significance of regularity at Figure 18.2 was evaluated with the help of PT1 at level $p < 0.0005$.

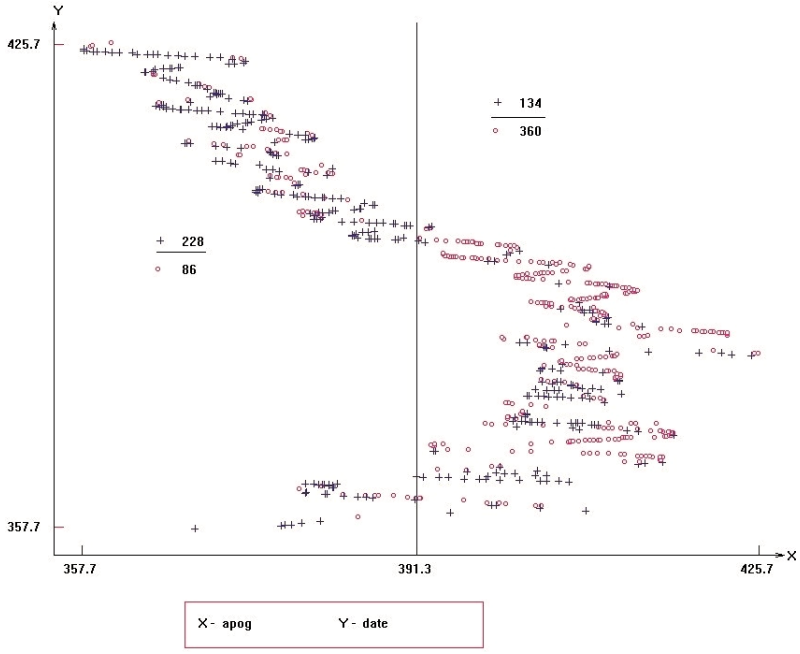


Figure 18.2. Sparse diagram illustrates influence of apogee altitude on the daily dose rate at ISS. Along the X axis apogee altitude is laid off and along the Y axis successive numbers of the days of measurements are given, $p < 0.0005$.

18.3 Redundancy Problem and Complete Verification Concept

The problem of the redundancy of discovered more complicated regularities arises when optimal partitions are searched inside more complex families [14, 16]. The problem occurs if verification is carried out using a simple version of the permutation test PT1. The problem consists in the existence of such patterns, valid in the sense of PT1, from the more complex two-dimensional Families III and IV, for which validity is actually determined by only one of the two X variables. Inclusion in their description of the second X variable is randomness only.

18.3.1 *Redundancy of two-dimensional regularities systems*

To solve redundancy problem, the concept of complete verification was suggested [14]. Complete verification implies verifying if each element of regularity is really important. One way to implement complete verification is use of permutation tests generating random samples with the help of random permutations of positions relative to the fixed vectors of prognostic X variables. However, permutations were allowed only within subsamples of the original training set, which were induced by one-dimensional partitions corresponding to simple patterns with a single boundary point.

However, this approach cannot be used for the verification of two-dimensional patterns, if there are no simplest valid one-dimensional patterns for X variables that are relevant to two-dimensional pattern under verification [14, 15].

Furthermore, experiments with real samples showed that verification procedure based on use of boundaries of one-dimensional pattern sometimes results in erroneous estimation of validity of the contribution of some variable X' into a two-dimensional pattern with relevant variables X' and X'', even when for X'' there is a valid one-dimensional pattern from Family I.

This effect occurs in case of significant deviation of threshold boundaries for X'' in two-dimensional partition from Family III and in one-dimensional pattern from Family I.

18.3.2 *Modified verification procedure*

Suppose that functional F_{int} is chosen. The algorithm for evaluating p-values of optimal two-dimensional partition at dataset $\tilde{S}_t = \{(y_1, x_1), \ldots, (y_m, x_m)\}$ includes the following steps:

(1) Find at $\tilde{S}_t$ optimal partition of two-dimensional admissible area for variables X_1 and X_2 with boundary points b_t^{o1} and b_t^{o2} correspondingly. Value $F_{\text{int}}(\tilde{S}_t, b_t^{o1}, b_t^{o2})$ is used as observed value of test statistics for both criteria: (a) test evaluating significance of X_1 and (b) test evaluating significance of X_2.
Test evaluating significance of X_1,

(2) Split $\tilde{S}_t$ in two subsets $\tilde{S}_t^l = \{(y_i, \mathbf{x}_i) | x_{i2} < b_t^{o2}, i = 1, \ldots, m\}$ and $\tilde{S}_t^r = \{(y_i, \mathbf{x}_i) | x_{i2} > b_t^{o2}, i = 1, \ldots, m\}$. Let m_l and m_r are numbers of objects in $\tilde{S}_t^l$ and $\tilde{S}_t^r$ correspondingly.

(3) Let $\tilde{\mathbf{S}}_{\text{tr}} = \{\tilde{S}_{\text{tr}}^{lj}, \tilde{S}_{\text{tr}}^{rj} | j = 1, \ldots, \text{N}\}$ where $\tilde{S}_{\text{tr}}^{lj}, \tilde{S}_{\text{tr}}^{rj}$ are received from $\tilde{S}_{\text{tr}}^l$ and $\tilde{S}_t^r$ by random permutations of target. This is obtained by considering two sets of N independent random permutations: set $\{\mathbf{f}_j^l = (f_1^{lj}, \ldots, f_m^{lj}) | j = 1, \ldots, \text{N}\}$ containing permutations of $\tilde{S}_{\text{tr}}^l$ object labels $(1, \ldots, m_l)$; set $\{\mathbf{f}_j^r = (f_1^{rj}, \ldots, f_m^{rj}) | j = 1, \ldots, \text{N}\}$ containing permutations of $\tilde{S}_{\text{tr}}^r$ object labels $(1, \ldots, m_r)$. For each $j \in \{1, \ldots, \text{N}\}$, $\tilde{S}_{\text{tr}}^{jl} = \{(y_{f_1^{lj}}, x_1), \ldots, (y_{f_m^{lj}}, x_{m_l})\}$ and $\tilde{S}_{\text{tr}}^{jr} = \{(y_{f_1^{rj}}, x_1), \ldots, (y_{f_m^{rj}}, x_{m_r})$.

(4) For each $j \in \{1, \ldots, \text{N}\}$, find on $\tilde{S}_{\text{tr}}^{lj} \cup \tilde{S}_{\text{tr}}^{rj}$ optimal partition of two-dimensional admissible area for variables X_1 and X_2 with boundary points b_{jr}^{o1} and b_t^{o2} and calculate $F_{\text{int}}(\tilde{S}_{\text{tr}}^{lj} \cup \tilde{S}_{\text{tr}}^{rj}, b_{jr}^{o1}, b_{jr}^{o2})$.

(5) Statistical significance of variable X_1 for two-dimensional regularity $\hat{p}$ is estimated as the proportion of datasets in $\tilde{\mathbf{S}}_{\text{tr}}$ where quality of partitions is better than in true dataset

$$\hat{p}_2 = \frac{\#\{\tilde{S}_{ir}^j | F_{\text{int}}(\tilde{S}_{\text{tr}}^{lj} \cup \tilde{S}_{\text{tr}}^{rj}) \geq F_{\text{int}}(\tilde{S}_t),\ j = 1, \ldots, \mathbf{N}\}}{\mathbf{N}}. \quad (18.2)$$

Test evaluating significance of X_2,

(6) Split $\tilde{S}_t$ in two subsets $\tilde{S}_t^l = \{(y_i, \mathbf{x}_i) | x_{i1} < b_t^{o1}, i = 1, \ldots, m\}$ and $\tilde{S}_t^r = \{(y_i, \mathbf{x}_i) | x_{i1} > b_t^{o1}, i = 1, \ldots, m\}$. Let m_l and m_r are number of objects in $\tilde{S}_t^l$ and $\tilde{S}_t^r$ correspondingly.

(7) Repeat steps 3 and 4 for sets $\tilde{S}_t^l$ and $\tilde{S}_t^r$.

(8) Statistical significance of variable X_2 for two-dimensional regularity is evaluated by p-value estimate $\hat{p}$ that is calculated as the proportion of datasets in $\tilde{\mathbf{S}}_{\text{tr}}$ where the quality of partitions is better than in true dataset:

$$\hat{p}_2 = \frac{\#\{\tilde{S}_{ir}^j | F_{\text{int}}(\tilde{S}_{\text{tr}}^{lj} \cup \tilde{S}_{\text{tr}}^{rj}) \geq F_{\text{int}}(\tilde{S}_t),\ j = 1, \ldots, \mathbf{N}\}}{\mathbf{N}}. \quad (18.3)$$

The represented procedure where p-values are calculated by formulas (18.2) or (18.3) will be further referred to as PT2. For functional F_{loc} all calculations are the same.

18.3.3 *Examples of two-dimensional regularities*

In the following, there is an example of using the OVP method for studying patterns associated with the occurrence of ischemic stroke (IS). in groups of patients with severe neurological disorders [9].

Figure 18.3 shows a two-dimensional pattern from Family III, which connects the occurrence of IS with the presence/absence in medical history of transient ischemic attack (TIA) and level of high-density lipoproteins (HDL-C). In group without TIA, the common tendency exists: IS occurrence is higher when HDL-C level is less than 0.55 mmol/l. On the contrary, IS occurrence is higher when HDL-C level is greater 0.55 mmol/l in group with TIA. The significances of contributions of TIA and HDL_C were estimated using PT2 at levels $p1 < 0.0005, p2 < 0.0005$.

Another regularity describes the relationship between vascular endothelial gross factor (VEGF) and the combination of oxygen saturation index (sO_2) and S100 proteins level in serum. VEGF is a signal protein involved in the process of blood vessels' formation. It

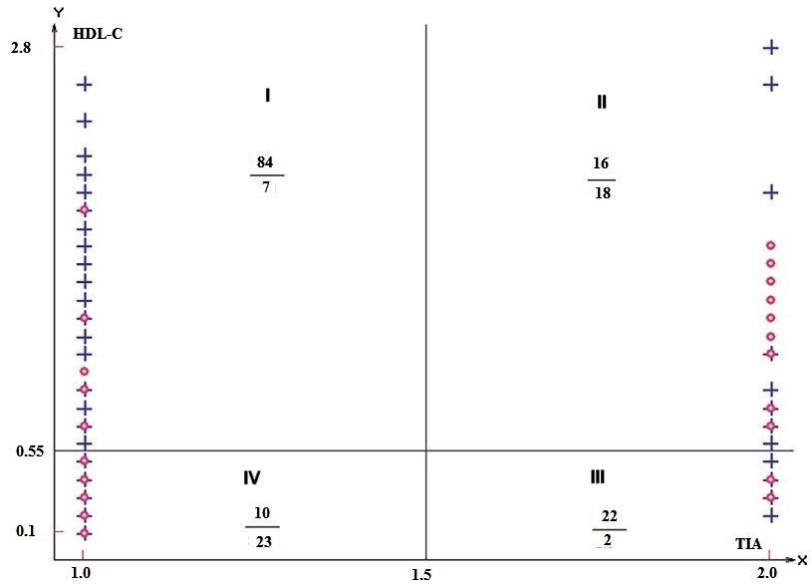


Figure 18.3. Scatter diagram that relates presence "o" or absence "+" of IS with the presence of TIA in medical history (X axis, 1.0 — absent, 2.0 — present) and with cholesterol level of high-density lipoproteins (HDL-C, mmol/L) (Y axis). Statistical significance of binary variable indicating cases with TIA was evaluated by PT2 as $p1 < 0.0005$, significance of HDL-C was evaluated as $p2 < 0.0005$.

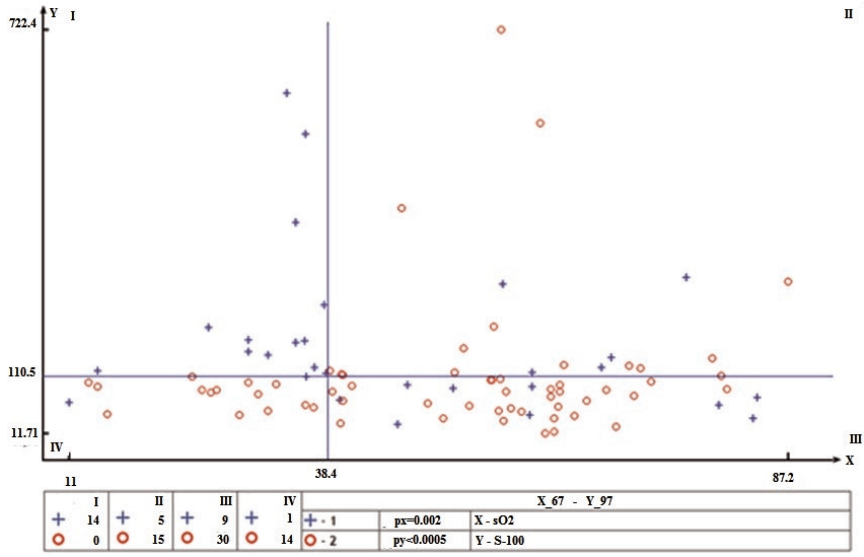


Figure 18.4. Association of high (>750 pg/ml) VEGF levels with content of S100 proteins in blood serum and oxygen saturation index sO_2 (%). The axes X and Y, respectively, show sO_2 (%) and S100 proteins' levels (ng/L). The cases with high VEGF are marked as "+.". The cases with low (<750 pg/ml) are marked as "o." Significance is evaluated as $p = 0.002$ for sO_2 and as $p = 0.0005$ for S100.

was shown by many researches that VEGF production is increased in patients with hypoxia. At that hypoxia is strongly related to low values of oxygen saturation factor sO_2 in venous blood. Regularity from Figure 18.4 shows that VEGF activation may be caused by the combination of low values of sO_2 and high values of S100.

18.4 Multiple Testing Effect

Estimates of p-values calculated by PT1 and PT2 evaluate the significance of individual one-dimensional patterns without taking into account a total number of evaluated indicators. However, in case of a large number of analyzed parameters, it is possible that significant differences between compared groups can occur incidentally for at least one of them even if these groups are identically distributed. The Bonferroni correction can be used to ensure the significance of the pattern at any given level [4]. Let us suppose that n is number of analyzed indicators. The pattern with the calculated p-value is

considered to be significant at level α if inequality $p < \frac{\alpha}{n}$ obviously equivalent to inequality $pn < \alpha$ holds true.

Note that using two-dimensional models requires even stronger multiple testing correction of statistical significance to compare with the correction necessary for one-dimensional models. The latter is associated with the need to calculate p-values for a large number of pairs of indicators. If the verification of one-dimensional models requires testing of n null hypotheses, where n is the number of indicators, then in order to verify all kinds of two-dimensional patterns, it is necessary to test $\frac{n(n-1)}{2}$ hypotheses. For example, in tasks with 50 initial supposed explanatory variables, it is necessary to test 1225 pairs.

In order to achieve the Bonferroni-corrected significance at a level of 0.01, it is necessary that the original significance be no worse than $p < 0.000005$. Therefore, Bonferroni correction is often not possible because of very strict requirements on initial value. However, Bonferroni correction is actually based on quite rough upper estimates for error probability of the first kind and appears to be very conservative in the sense of unjustified rejection of actually important patterns [3, 12]. The approach based on a comparison of a set of significant patterns found in a real sample with a set of patterns found in a random sample seems to be more effective. In this case, random samples are generated using the same procedure of random permutation of positions of target variable as the one used in permutation tests. The use of permutation tests to account for the multiple-testing effect makes it possible to take into account the complex interaction of the test statistics [3]. A similar approach was used in [7] to implement multiple testing corrections in OVP analysis. The efficiency of this was evaluated in a task where two groups of countries with two different systems of basic social institutions.

18.5 Comparing of Regularities in Two Independent Samples

Previously we considered methods of construction and verification of individual patterns. Often, however, the question arises about how significantly patterns differ between two groups of objects. For example, when evaluating the effect of some binary factor on the biological process under study, it is useful to estimate the differences

between the patterns that exist in groups induced by this binary factor. Thus, it is of unconditional interest to search for valid differences between the patterns in the groups with different variants of some gene.

18.5.1 *Functional for discrepancy evaluation*

Previously, the method was briefly described in [9, 16]. To assess the validity of differences between patterns, a new variant of the OVP method, which uses a modification of the permutation test, was proposed. Let us consider first the case of verification of the simplest one-dimensional pattern that describes dependence of Y on some variable X_i. Let us suppose that we need to compare the patterns that exist in the groups of objects

$$\tilde{S}_t^1 = \{s_1^1 = (y_1^1, x_1^1), \ldots, s_{m1}^1 = (y_{m1}^1, x_{m1}^1)\} \quad \text{and}$$
$$\tilde{S}_t^2 = \{s_1^2 = (y_1^2, x_1^2), \ldots, s_{m2}^2 = (y_{m2}^2, \mathbf{x}_{m2}^2)\}.$$

Let us first consider verification for the simplest one-dimensional Family I. Let us construct an optimal partition of the domain of values $\mathbf{M}$ by variable X_i within Family I using the procedures described in the previous sections. Let us suppose that we obtained the optimal partition $\{q_1, \ldots, q_r\}$, which in turn induces the partitioning of group $\tilde{S}_t^1$ into subgroups $\{\tilde{s}_1^1, \ldots, \tilde{s}_r^1\}$ and group $\tilde{S}_t^2$ into subgroups $\{\tilde{s}_1^2, \ldots, \tilde{s}_r^2\}$. Let $\hat{y}(\tilde{s}_j^k)$ be the average value of the predicted value Y for the subgroup $\tilde{s}_j^k$. For quantitative estimation of the differences between relationships of Y with the variable X_i in the groups $\tilde{S}_t^1$ and $\tilde{S}_t^2$, one can use a functional of comparison

$$F_{\text{int}}^c(\tilde{S}_t, \tilde{q}) = \sum_{i=1}^{r} \left[\tilde{y}(\tilde{s}_j^1) - \tilde{y}(\tilde{s}_j^2)\right] \sqrt{m_i^1 m_i^2}, \tag{18.4}$$

where m_i^1, m_i^2 are the numbers of objects in the groups $\tilde{s}_j^1$, $\tilde{s}_j^2$. To estimate statistical validity of differences described by functional F_{int}^c the following versions of the permutation test are used

18.5.2 *One-dimensional regularities*

The procedure for calculating the validity of the difference between one-dimensional regularities with one boundary is represented in the

following:

(1) Find on union $\tilde{S}_t^1, \tilde{S}_t^2$, optimal partition of interval (a_l, a_r) with boundary point b_t^o. Value $F_{\text{int}}^c(\tilde{S}_t^1, \tilde{S}_t^2, b_t^o)$ from (18.4) is used as the observed value of test statistics.
(2) Take set $\tilde{\mathbf{S}}_{\text{tr}}^2 = \{\tilde{S}_{\text{tr}}^{j1}, \tilde{S}_{\text{tr}}^{j2} | j = 1, \ldots, \text{N}\}$ where $\tilde{S}_{ir}^{j1}$, $\tilde{S}_{ir}^{j2}$ are random permutations of $\tilde{S}_t^1$ and $\tilde{S}_t^2$ correspondingly. This is obtained by considering N independent random permutations of object labels $(1, \ldots, m_1)$ and $(1, \ldots, m_2)$ as was done in previous sections. For each $j \in \{1, \ldots, \text{N}\}$ $\tilde{S}_{ir}^{j1} = \{(y_{f_1^j}, x_1), \ldots, (y_{f_m^j}, x_{m_1})\}$, $\tilde{S}_{ir}^{j2} = \{(y_{g_1^j}, x_1), \ldots, (y_{g_{m_2}^j}, x_{m_2})\}$.
(3) For each $j \in \{1, \ldots, \text{N}\}$ find on $\tilde{S}_{ir}^{j1} \cup \tilde{S}_{ir}^{j2}$ optimal partition of interval (a_l, a_r) with boundary point b_{jr}^o and calculate $F_{\text{int}}^c(\tilde{S}_{\text{tr}}^{j1}, \tilde{S}_{\text{tr}}^{j2}, b_{jr}^o)$.
(4) The estimate of p-value $\hat{p}$ is estimated as a proportion of datasets pairs in $\tilde{\mathbf{S}}_{\text{tr}}^p$ where the quality of partitions is better than in the true dataset:

$$\hat{p} = \frac{\#\{\tilde{S}_{ir}^{j1}, \tilde{S}_{ir}^{j2} | F_{\text{int}}^c(\tilde{S}_{\text{tr}}^{j1}, \tilde{S}_{\text{tr}}^{j2}, b_{jr}^o) \geq F_{\text{int}}^c(\tilde{S}_t^1, \tilde{S}_t^2, b_t^o),\ j = 1, \ldots, \mathbf{N}\}}{\mathbf{N}}$$

The represented procedure will be further referred to as PT3.

18.5.3 *Example of regularities associated with difference between one-dimensional optimal partitions in two groups*

Below there is an example of using the modification, described in this section, of the OVP method for studying the patterns associated with the occurrence of ischemic stroke (IS). The left diagram from Figure 18.5 shows a one-dimensional pattern from Family I, which connects the occurrence of IS with glucose level in a group of patients without transient ischemic attack (TIA). The right diagram from Figure 18.5 shows one-dimensional pattern from Family I, which connects the occurrence of IS with glucose level in group of patients with transient ischemic attack (TIA). There is no evident relationship between glucose levels in a group without TIA: IS occurrence rates are close when glucose concentration is higher 6.3 mmol/L or

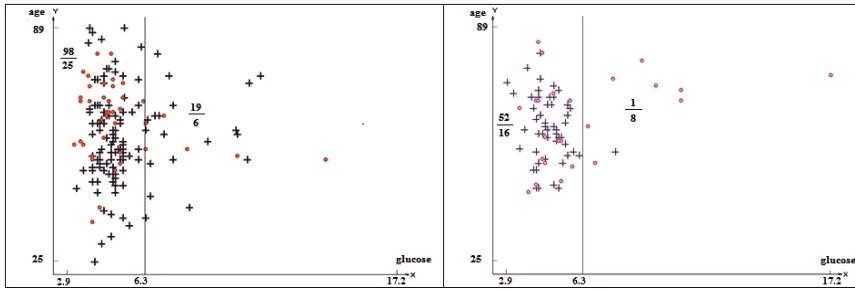


Figure 18.5. Association of IS with glucose content in blood of patients, depending on whether they had TIA. The axes X and Y, respectively, show levels of glucose (mmol/L) and age (years). The left scatter diagram corresponds to the cases without TIA. The right scatter diagram corresponds to the cases with TIA. The cases of IS are marked as "o." Cases without IS are marked as "+." $p = 0.001$.

when it is below this threshold. On the contrary, in a group with TIA, there is only one case without stroke when glucose concentration is higher 6.3 mmol/L against 8 cases with IS. Statistical significance of the observed effect was evaluated at level 0.001. At that, there are 52 cases without ischemic stroke and only 16 cases with stroke when the glucose level is below the threshold

18.6 Optimal Valid Partitioning Technique for Searching of Cyclic Variations in Time Series

Optimal valid partitioning method may be used as a tool for the analysis of cyclic variations. Let J be a variable denoting the number of the observation from the initial point and p be a presupposed cycle length. The cycling variable C_p is the quantity that describes the phase of a cycle with period p. The value C_p for the day number J is calculated as the residue of division of J by p, or $C_p(J) = J/p - [J/p]$.

Obviously, the cycling variable C_p takes the values in the segment [0, 1]. One may also use the segment [0, 100]. If the time series has a periodic alternation of time intervals with high and low values of the predicted quantity with a cycle length p, an interval in the segment of values of cycling variable C_p is selected in which Y values significantly differ from Y values in the remaining part of the interval. Such regularities in data are revealed by the method of optimal partition using

the model with two boundary points. The method of optimal partition enables one to estimate statistical significance of the revealed regularity, which can be considered as statistical significance of the corresponding cyclic variations. The discussed technique was used to reveal radiation oscillation at the International Space Station [17] when the apogee altitude is less 391.3 km (see Figure 18.2).

All cycling variables from C_7 to C_{107} were tested for this purpose. The most significant regularity was found for cycling variable C_{60}. This regularity is represented in Figure 18.6. The sign "o" marks the days when the level of 17.3 mrad was exceeded and the sign "+" marks the days with the level below 12.6. The two end regions correspond to higher daily doses, and the central one corresponds to lower values. Statistical validity of regularity from Figure 18.6 was evaluated by PT1 at level $p < 0.005$. Existence of regularity from

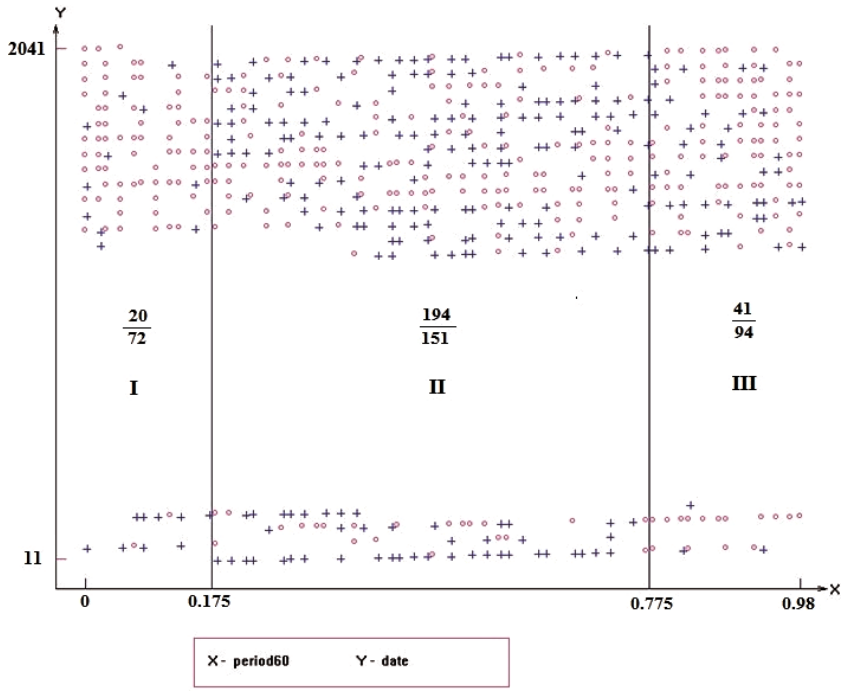


Figure 18.6. The influence of cyclic oscillations of radiation level with a period of 60 days when apogee altitude is less 391.3 km. Along the X cycling variable C_{60} is laid off and along the Y axis successive numbers of the days of measurements are given, $p < 0.0005$.

Figure 18.6 is related probably to nodal precession of ISS orbit with a period of two months. High radiation level corresponds to precession phase when the orbit passes closely to South Atlantic Geomagnetic Anomaly.

18.7 Conclusion

The discussed optimal valid partitioning method potentially may be used in different areas of data analysis. OVP technique has important advantages over some alternative types of statistical analysis. The first advantage consists of the possibility to verify regularities correctly by the same dataset where optimal boundary points are searched. It is achieved by recalculating boundaries at every random dataset with a permuted target. Another advantage is convenient for the user visual form of regularities presentation. The method allows to reveal and to describe joint effects of pairs of factors on outcome with evaluating statistically if each factor is really necessary. In high-dimensional tasks, it is possible to implement corresponding multiple testing corrections.

Acknowledgments

This study was funded by the theme 0063-2019-0003.

References

[1] Abdolell, M., LeBlanc, M., Stephens, D., and Harrison, R.V. (2002) Binary partitioning for continuous longitudinaldata: Categorizing a prognostic variable. *Statistics in Medicine*, 21, 3395–3409.

[2] Breiman, L., Friedman, J.H., Olshen, R.A., and Stone, C.J. (1984) *Classifcation and Regression Trees* (Chapman and Hall, USA).

[3] Dudoit, S., Popper Shaffer, J., and Boldrick, J.C. (2003) Multiple hypothesis testing in microarray experiments. *Statistical Science*, 18, 71–103.

[4] Dunn, O.J. (1961) Multiple comparisons among means. *Journal of the American Statistical Association*, 56, 52–64.

[5] Good, P. (2005) *Permutation, Parametric and Bootstrap Tests of Hypotheses*, 3rd Ed. (Springer Verlag, New York).

[6] Ernst, M.J. (2004) Permutation methods: A basis for exact inference. *Statistical Science*, 19, 676–685.

[7] Kirilyuk, I.L., Kuznetsova, A.V., Sen'ko, O.V., and Morozov, A.M. (2017) Method for detecting significant patterns in panel data analysis. *Pattern Recognition and Image Analysis*, 27, 94–104.

[8] Kuznetsov, V.A., Ivshina, A.V., Senko, O.V., and Kuznetsova, A.V. (1996) Syndrome approach for computer recognition of fuzzy systems and its application to immunological diagnostic and prognosis of human cancer. *Mathematical and Computer Modelling*, 23, 92–112

[9] Kuznetsova, A.V., Kostomarova, I.V., and Senko, O.V. (2014) Modification of the method of optimal valid partitioning for comparison of patterns related to the occurrence of ischemic stroke in two groups of patients. *Pattern Recognition and Image Analysis*, 24, 114–123.

[10] Lubinsky, D. (1994) Classification trees with bivariate splits. *Journal of Applied Intelligence*, 4, 283–296.

[11] Pak, C.S. and Shaun, M.P. (2014) Statistical power and significance testing in large-scale genetic studies. *Nature Reviews Genetics*, 15, 335–346.

[12] Pesarin, F. and Salmaso, L. (2010) *Permutation Tests for Complex Data. Theory, Applications and Software* (John Wiley & Sons, Ltd., UK).

[13] Senko, O.V. and Kuznetsova, A.V. (1998) The use of partitions constructions for stochastic dependencies approximation. *Proceedings of International Conference on Systems and Signals in Intelligent Technologies, ICSSIT*, pp. 291–297.

[14] Senko, O.V. and Kuznetsova, A.V. (2006) The Optimal ValidPartitioning Procedures. InterStat, Statistics in Internet. http://ip.statjournals.net:2002/InterStat/.

[15] Senko, O. and Kuznetsova, A. (2010) A recognition method based on collective decision making using systems of regularities of various types. *Pattern Recognition and Image Analysis*, 20, 152–162.

[16] Senko O.V. and Kuznetsova, A.V. (2011) Methods for discovering and analysis of regularities systems. Approach based on optimal partitioning of explanatory variables space. *Proceedings of International Conference on Knowledge Discovery and Information Retrieval, KDIR*, pp. 423–426.

[17] Tsetlin, V.V., Nosovskii, A.M., Senko, O.V., Kuznetsova, A.V., and Ol'shanskaya, E.V. (2010) A patten recognition method for short-term forecast of the radiation situation in long term manned space stations. *Pattern Recognition and Image Analysis*, 20, 396–403.

[18] Zhuravlev, Y.I. and Ryazanov, V.V. (2016) Search, processing and application of logical regularities of classes. *Information Technologies & Knowledge*, 10, 216–232.

© 2025 World Scientific Publishing Company
https://doi.org/10.1142/9789811267215_00019

Chapter 19

Mathematical Methods of the Blood Vessels Analysis on the Biomedical Images

Natalya Yu. Ilyasova*, Nikita S. Demin†, and Rustam A. Paringer‡

Image Processing Systems Institute, NRC "Kurchatov Institute"
Samara National Research University, Samara,
the Russian Federation
**ilyasova.nata@gmail.com*
†*volfgunus@gmail.com*
‡*apathyntg@gmail.com*

In this chapter, information technology of the analysis of biomedical diagnostic images, containing dendritic structures, in particular, images of fundus blood vessels, and crystallograms of biological liquids is considered. The mathematical apparatus is presented for estimation of numerical characteristics (diagnostic features) on the basis of the geometrical parameters estimation of the medical diagnostic images, allowing to automate diagnostic researches and to raise reliability of medical diagnostics. Mathematical models of objects with correspondence to various classes of diagnostic images are developed such as dendritic structures. The mathematical methods used in the chapter include the method of local fan-beam transformation, the textural analysis, the wavelet analysis and the discriminant analysis. Experimental researches

are presented upon modeled images as well as upon natural diagnostic images.

Keywords: Medical diagnostic systems, information technologies, monitoring programs, diabetic retinopathy, hypertensive retinopathy, optic nerve, image processing, digital analysis, morphological features, disease detection, diagnostic information, diagnostic criteria, vessel separation, computer system, hardware/software complex, diagnostic devices, AlluraXper system, pathologic vessel changes, noise, eye fundus images, microvessel changes, diagnostic features, geometrical characteristics, clinical investigations, radial functions, discriminant analysis.

19.1 Introduction

Computer analysis of images became the basic tool of medical diagnostic systems allowing one to increase diagnostics quality significantly [11,12,24,37,39,40,43,44]. Information technologies are most widely introduced in ophthalmology and cardiology [26,38]. Modern medicine is one of the most high-technology industries the most important task of which is to develop new and efficient methods for early diagnostics of various health problems. The last few decades have been characterized by a significant breakthrough in the field of technological infrastructure of medicine. For the time being almost all examination techniques in ophthalmology, cardiology, and other areas of medicine have been computerized. We analyze images of vascular systems of eye fundus and coronary vessels. In both cases, the blood vessel images carry important diagnostic information.

As for ophthalmology, with the appearance of digital fundus cameras and then laser scanning ophthalmoscopes, fundus imaging may be obtained and printed out within a few seconds. The human fundus imaging system contains information about vascular, ophthalmic, and even systemic diseases such as diabetes, hypertonia, and arteriosclerosis. Except for the usage in large monitoring programs, the digital image analysis of the human vascular system is capable of helping in assessing the severity of diabetic retinopathy [17], age-related retinopathy and retinopathy of prematurity [5], as well as in finding a foveal avascular zone [14], narrowing of arterioles and measuring a diameter within the framework of hypertonia diagnostics, to define relationships between vascular tortuosity and hypertensive

retinopathy [8], in early detection of such diseases as degeneration of the optic nerve and the retinal separation. Analysis of changes in characteristics of blood vessels and various diseases contributes to early disease detection and treatment still in the early stages.

Computer-aided analysis is also used in computerized laser surgery and for biometric identification [29, 38]. In addition to the development of biomedical computer technologies and analysis and image processing systems, the semi-automatic and later fully automated computerized systems of recognition and quantitative analysis of microvascular changes turned out to appear. Creating of digital analysis technique for the human vascular system (VS) is impossible without solving a number of issues, i.e. detection of blood vessels in digital images, selection of methodology for the blood vessel analysis, and a computing model of a vascular bed for subsequent calculations, and determination of standard values for diagnostic parameters. Detection of blood vessels and determination of their morphological features are the key stages of computerized methodologies of the diagnostic analysis of the human vascular system (VS), since the diagnostic results will be dependent on the accuracy of detection and measurement of its elements.

The goal of our study is as follows: Developing of a methodology allowing one to get from the vessel images important diagnostic information, developing and examination of methods, algorithms, and information technologies of modeling, processing, and analysis of blood vessel images, as well as creation of computer systems for medical purposes on the basis of the obtained results. Such systems have to provide solving of early and differentiated vascular disease detection problems with the aid of quantitation of morphological characteristics. For this purpose, we solved the following problems:

(1) We analyzed the up-to-date state of the problem of vascular pathologies detection, revealed the main steps of blood vessel image processing, and determined informative characteristics for clinical recognition.
(2) We worked out a methodology of extracting the blood vessel images' important diagnostic information, which includes mathematical models, methods, and computer systems.
(3) We developed a mathematical model of blood vessels' imaging allowing one to provide a formal description of geometrical

parameters and to construct diagnostic criteria. We also developed methods and algorithms for enhancing accuracy in the estimation of geometric parameters.

(4) On the basis of quantitation of changes in vascular system states and methods of estimation of the corresponding diagnostic criteria, we proposed a set of integral indicators of the vascular system state. We analyzed their efficiency and proposed methods of extracting important diagnostic information about the state of an individual vascular system.
(5) We developed information technology for the analysis of images of eye fundus vessels including methods of generation of the space of important signs for the effective early detection and prophylaxis of development of the diabetic retinopathy in patients with insular diabetes.
(6) We proposed information technology for the restoration of spatial structure of coronary vessels from a small number of unmatched angiographic projections that were oriented toward the estimation of local spatial geometrical characteristics of vascular system and diagnostic parameters.
(7) We worked out guidelines, algorithms, and software for diagnostic systems supporting ophthalmologists and cardiologists in decision-making process.

19.2 Computer Systems for Geometrical Analysis of Blood Vessels Diagnostic Images

The promising research direction is the development of software with different auxiliary algorithms for the analysis of images of coronary and retinal vessels including the vessel separation, determination of the important image elements, the measurements of vessel diameters, vessel tortuosity angles, and other diagnostic signs of vascular systems. All these types of software can quicken the study of relations between changes in the anatomy of the human vascular systems and different diseases.

The SIRIUS system (System for the Integration of Retinal Images Understanding Services) [35] is a computer system for the analysis of retina images on the basis of the web application that provides a possibility of collaborative work for ophthalmologists and other

experts in the field. The SIRIUS system consists of web interface, and an application server that gives access to different services and tools for image processing. The system allows exchange of images and provides different computer-aided methods for reducing errors when analyzing retina images. The system service module for the analysis of vessel microcirculation includes semiautomatic algorithm for the calculation of arteriovenous ratio (AVR). The weakness of this system is the absence of the possibility to measure different diagnostic features of vessel pathological changes when detecting pathologies on the retina images.

A longstanding competition known as ROC (Retinopathy Online Challenge) [34] is focused on determination of the best algorithms for different aspects of diabetic retinopathy detecting. It allows one to cooperate when developing methods for creating a high-performance computer system for solving monitoring and diagnostic problems [31]. The project RIVERS (Retinal Image Vessel Extraction and Registration System) [41,42] also makes efforts in the same direction. The software application VAMPIRE (Vascular Assessment and Measurement Platform for Images of the Retina) [36] allows one to carry out semi-automatic estimation of retina vessels and their characteristics. This computer system serves for efficient and reliable detecting of the most important retinal elements (optic disc, vasculature) and estimation of the key parameters used frequently in investigative studies: vessel diameter, tortuosity, and vessel branching angles.

A lot of large companies and research centers developed their own computer systems for automatic processing and analysis of medical images for special- and general-purposes. Each of the systems has its own distinctive features, merits, and demerits. The system "Gamma Multivox" [13] consisting of a hardware/software complex for data inputting, processing, and storage was developed in Lomonosov Moscow State University. The complex includes automated workstations designed for multimodal operation with 2D/3D medical images of lungs, bronchus cavities, and different anatomical units (teeth, paranasal sinuses, contrasted vessels inside soft tissues, and so on), in particular, with coronary vessel images. The main part of the workstation is a special-purpose software that allows a doctor to see images obtained with the aid of medical diagnostic devices, which use different physical registration methods. The system "Gamma Multivox" allows one to visualize and to process 2D/3D

images. It makes it possible to work with several 2D/3D images and several series of images, as well as synthesis, visualization and processing of 3D vessel images.

The Delante company offers an alternative diagnostic software for analysis of medical images including vessels [33]. They developed two high-end applications for diagnostic workstations IQ-VIEW and Myrian. The IQ-VIEW program can integrate with any image transmitting and archiving system (PACS). It unites all functions of a full-scale workstation for two-dimensional images reading and a workstation for 3D post-processing and realizes preprocessing of any data from three-dimensional images. The analogous requirements to system resources characterize the diagnostic software Myrian. One of its moduli is the Myrian® XP-Vessel application that simultaneously supports two modalities: vessel visualization by computed tomography method and by nuclear magnetic resonance method. This is the most suitable offer of the company for vessel monitoring. The Myrian® XP-Vessel distinguishing features are automatic extraction of the central line, smoothed presentation, measuring of the vessel diameter and length, curved planar reconstruction (CPR), and axial visualization (in transversal sections). Moreover, this application creates a diagram of the relation between the vessel diameter and its length, as well as shows the vessel cross-sections. In addition, the application provides an automatic segmentation of vessels as well as the calculation of their central lines. All these possibilities combined with other advantages of the system make it suitable for vessel images post-processing. However, it does not allow one to calculate vessel diagnostic signs for revealing their deformations of different kinds and the pathology degree.

Of all the present systems, the best one is the AlluraXper system [2], produced by Philips company. It allows one to carry out a range of interventional cardiology and vessel procedures. The Allura 3D system ensures three-dimensional visualization of the brain, cerebral, peripheral vasculature, and bone structures and performs three-dimensional visualization based on information obtained from a single rotational angiography run of coronary artery minimizing the effects of perspective distortions. The facilities of the AlluraXper system that are compatible with the DICOM 3.0 standard provide suitable federation with other systems and guarantee the availability of informational images when it is necessary. The optimized federation

of the system combined with a large number of built-in options, which allows one to construct correct three-dimensional vessel models, to eliminate noise in real time, and so on, is one of many advantages of the system developed by Philips. However, unfortunately, the system is highly specialized and it costs a lot.

One more up-to-date computer system of image post-processing that is not connected tightly with the equipment, was proposed by the Mediso company (Medical Imaging Systems, Budapest, Hungary). Their InterView Fusion product [25] is the system for visualization, processing and analysis of medical vessel images of different origins. The InterView Fusion system allows one to download and superpose images with the aid of specialized viewer and automatic registration algorithms, and to carry out static measurements over the regions and volumes of the object of interest. The system is designed for processing of images of different types. It does not have special tools for the analysis of vessel images, but only the general set of the supplied packages.

Today, in many countries in public screening centers, they heavily use quantitative approaches of estimations of vessel images when revealing vascular pathologies with the aid of automated systems of pattern recognition (Goldbaum, Taylor, Abramoff, Kelvin, Perez-Rovira, and Stewart). However, our examination of the currently existing software complexes for the analysis of vascular system images showed that most of them do not have applied software for measuring the full set of diagnostic features and diagnosis. They have only tools for image registration, record-keeping diagnostic information about patients, and the most frequently used tools for pre-processing, image quality improvement, and marking the images. Therefore, the urgent problem is the development of a system for analysis of subclinical morphological changes allowing one to automatize diagnostic stages and to carry out quantitative monitoring of pathologic vessel changes.

Diagnostic software complex "OphthalmOffice" for analysis of the vessel system of eye fundus: We developed a method for diagnostics of early stages of diabetic retinopathy and a methodology of quantitative estimation of elements of pathomorphology picture of eye fundus physicians use when defining a degree of retina pathology. Our method allows one to achieve higher measuring accuracy of diagnostic signs accustomed to the majority of physicians that

can correspond to the physician's own experience and the results of known clinical investigations [16].

The software complex is intended for the eye fundus vessels tracing, localization, and analysis of the region of the optic nerve disk, calculation of the basic set of geometrical characteristics of the eye fundus vessels, as well as calculation of diagnostic signs, classification of vessels and carrying out planimetric studies that include dynamic analysis of images. The functional specification of this complex includes entering and pre-processing of eye fundus images, quantitative estimation of the basic types of microvessel changes in diseases: the mean diameter alteration, the diameter irregularity, the vessel tortuosity, and so on. It also provides the objective control of dynamics of changes of the pathological region sizes on subsequent images, estimation of the degree of pathology, and automatic entering of database on patients, images, and measurements. As input data, it uses images in BMP, JPG, GIF, TIF, and PNG formats. This allows the user to obtain input data from practically any available device for digital registration of fundus images.

The diagnostic complex is divided logically into the following systems: the system that guarantees data storing and generates reports, the system of estimation of eye fundus and optic nerve disk diagnostic signs, the system for planimetric analysis, and the system for classification and diagnostic research. The system for classification and diagnostic examination provides the tools of the correlation and discriminant analyses for generating the space of more informative signs. For adjusting the classifier, it provides the tools for generating the optimal sampling with regard to the criterion of the efficiency of dividing into groups of pathologies [22]. When developing this complex: we worked out a new generalized mathematical model of two classes of blood vessels' images, namely eye fundus vessels and coronary vessels. This model allows one to formalize the description of geometrical characteristics and to determine diagnostic features of fundus vessels and coronary vessels. For vessel tracing and estimation of local directions, we worked out a new method of the modified local fan transform based on analysis of the luminance radial functions in the region of the sliding sector inside the scanning window (up to now a modification of the ray transform of the luminance function was used) [30]. We developed an information technology for generation of the space of effective features for analysis of eye fundus

vessels' images. Then based on the discriminant analysis we selected the most informative sets of diagnostic signs allowing us to diminish errors by up to 2.5% when sorting the vessels by groups normal and different stages of pathologies (insular diabetes). We solved the problem of selection of vessel segment central lines based on a modified wavelet transform by using undirected 2D wavelets formed from 1D ones according to the proposed model of vessels (other authors did not take into account vessel models, and this led to the selection of different artifacts together with the vessels). Our method showed high accuracy when selecting vessel central lines, high noise immunity (due to the wavelet smoothing properties), and invariance with regard to the vessel orientation and the vessel image luminance. The algorithm steady work is certified when the ratio noise/signal is less than 0.9 and at that, the values of the vessel trace standard deviation are not larger than 0.6 pixels (see Figure 19.1) [19].

The system of coronary vessels analysis: The proposed system is intended for solving numerical simulation problems, estimation of geometrical characteristics, and visualization of three-dimensional structure of the vessel tree based on the data of the angiographic study of the coronary vessels [18]. The functional specification of the complex includes data reading of x-ray angiographic studies of coronary vessels in DICOM v.3.0 format, modeling of the filming geometrical characteristics, neutralization of geometrical distortions and synchronization of projections, numerical simulation and estimation of geometrical parameters of the three-dimensional structure of the coronary vessels, and visualization of the spatial structure of the coronary vessels.

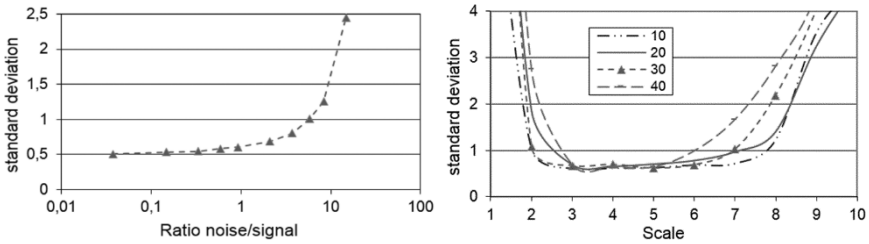


Figure 19.1. The dependence of the standard deviation of the central line: On the value of the ratio noise/signal (left), on the noise level for different scales (σ_{wym} changes from 10 to 40).

The methods and algorithms of processing and analyzing coronary vascular systems images developed under the investigation process made the basis of four subsystems of the program complex "CardiOffice": *the subsystem for input data processing*; *the preprocessing subsystem* (*it* accomplishes the time synchronization of the projections and geometrical compensation of the projections images); *the modeling subsystem* (it consists of two modules: the module for geometrical binding of projections and the tracing module, the subsystem realizes the modeling of the filming geometrical characteristics, performs data simulation and estimation of geometrical characteristics of the three-dimensional structure of the coronary vessels); *the visualization subsystem* (they realize the visualization of the spatial structure of coronary vessels using the tools of the OpenGL graphic library) (see Figure 19.2).

The data accumulation procedure for the following processing was carried out in hospitals TelHashomer and HADASA (Jerusalem) with the aid of special C-ARM equipment. The obtained cineangiography data included information about the cineangiography geometry, the projection films constituting frames that are angiographic images of coronary vessels made in fixed instants of time and under

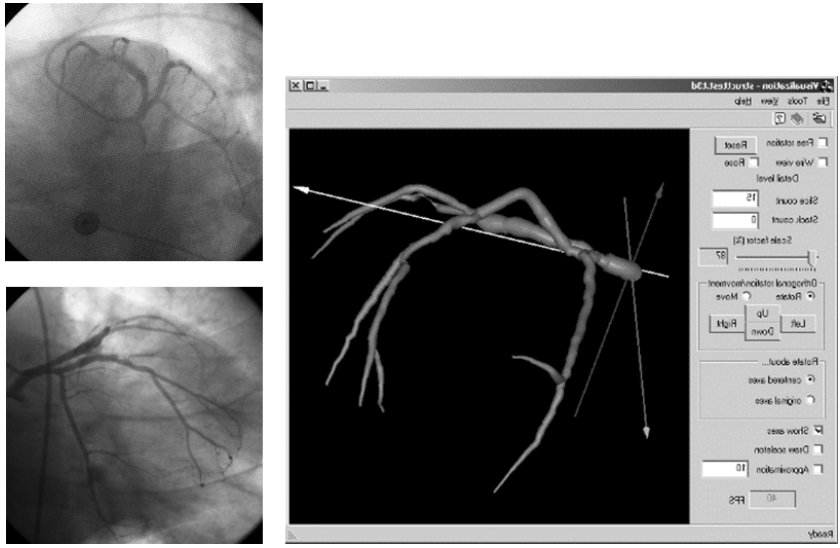


Figure 19.2. The restoration results: The initial projections of coronary vessels and graphical interface of the visualization subsystem.

a certain angle. They all were stored in a uniform medicine information storage format DICOM. On average, each separate document in DICOM format contains 4–6 films with images of the left coronary artery and 2–3 films with images of the right one. Each frame is an image in BMP (grayscale) format and its size is 512×512 pixels.

When developing the system we used information technology for restoration of three-dimensional structure of coronary vessels with the aid of a small number of angiographic projections. It allows one to estimate spatial geometrical characteristics of the coronary vessel system and to formulate diagnostic features under the conditions of unmatched projections. We solved the problem of restoration of the spatial structure of vessels using a small number of planar projections under the view. This problem belongs to the class of ill-posed problems and therefore it is extremely complex. The problem complexity increases since the object under investigation is a dynamic one, and therefore due to the non-simultaneous registration we, in fact, study projections that are unmatched in time. Besides, there is noise on the X-ray angiographic projections due to the use of low-energy X-radiation.

We propose the information technology of automatic restoration of the three-dimensional structure of the vascular system and estimation of its geometrical characteristics. The main method it uses is the simultaneous analysis of vessel images on the projections and the parallel restoration of the geometrical structure of the vascular tree. The specific feature of this technology is using of a small number of angiographic projections. It allows one to restore spatial structures of objects that most closely correspond to given projections from available noisy central projections of a dynamic object obtained in different instances of time.

19.3 Methods for Estimating Vessel Parameters

Detection of blood vessels and determination of their morphological features such as width, length, tortuosity, angles of bending vessels, etc., is now widely used for diagnostic purposes, monitoring, medical care, and evaluation of various cardiovascular, ophthalmic, and other diseases. Manual processing and analysis of diagnostic images (segmentation of blood vessels and measuring their width and tortuosity)

is a long-term and fatigue issue which also requires special training and professional skills [9]. Specialists have to learn to distinguish damages in the human vascular system (VS) from its visual appearance without pathologies. Moreover, a human subjective factor has always occurred in such kind of analysis that often results in differences in diagnosis determined by physicians. The medical community has agreed that automatic quantification of the human vascular system (VS) is the first step in the development of computerized diagnosis systems for human diseases. Many algorithms and methods of numerical analysis of the human vascular system have been published.

In a most quick way, the pathology can be detected by estimating the blood vessel width [3]. According to modern data, the deviation of the diameter of a pathological vessel from a normal one is just about 20%. This fact is a major motivation behind the development of the most precise methods for estimating the retinal arterioles. The vascular disease is diagnosed using diagnostic features based on the geometric parameters of vessels. Among the most important parameters are the vessel's diameter and direction.

Recent years have seen the development of many image-processing algorithms aimed at analyzing both the vascular system in general and the optic disk, in particular. The algorithms use different approaches and have their benefits and shortcomings [7, 19, 22, 30]. The algorithms based on a complex continuous wavelet transform [7, 19] are best suited for describing the structure of lines in different directions. A method for estimating vascular parameters proposed in [6] relies on mathematical morphology as an instrument for extracting image fragments suitable for the description of boundaries and skeleton of a vessel. The method utilizes a sparse representation of signals. In [6], each signal is assumed to be composed of a linear combination of several morphologically different components. The final map of a vessel is constructed using an adaptive threshold method. The method was shown to perform well in detecting anomalies and pathologies in the retina image. When used on their own, a major disadvantage of the morphological techniques is that they disregard the information on the vessel profile shape. Besides, while searching just for elongated structures, heavily twisted vessels can be missed.

A method for estimating the vessel diameter using an algorithm based on a parametric model of an arbitrarily complex-shaped vessel was reported in [28]. The automatic algorithm is capable of segmenting the entire vascular tree and calculating the vessel's diameter and direction in a digital ophthalmologic image. An algorithm that utilizes a new parametric surface model of the vessel intensity profile was described in [32]. Compared to other methods, the said method offers the advantage of robustness, whereas, as the authors mention, the negative side is the dependence of the results on the test data and experimental conditions. If vessel diameters in the test data vary in a wide range, the measurement accuracy significantly deteriorates. Other approaches to measuring the diameter of vessels rely upon the approximation of the vessel brightness profile. With the vessel brightness normally having a Gaussian profile, a Gaussian curve is often used to approximate the vessel cross-section. However, with the diseased vessels tending to have well-pronounced boundaries, their profile looks like a combination of two Gaussians, making it difficult to measure their diameter in an automatic mode and creating a possibility to falsely recognize a single vessel as two ones.

We propose a method for estimating local parameters of a fundus vascular system that exploits a modified fan transform, which is more noise-immune thanks to the additional filtering and noise averaging. Another advantage is that the method can perform a high-efficiency analysis of bifurcations, crossovers, and termination of vessels in the presence of interfering factors, such as close vessels.

19.3.1 *Development and study of methods for estimating retinal vessel parameters using a modified local fan transform*

We introduce a modified local fan transform (MLFT) intended to operate with real-life low-quality vascular network imagery, which is well suited for processing images characterized by spots and closely spaced vessels. With this method, the brightness distribution function is analyzed sector-wise depending on the sector's radius, size, and rotation angle (see Figure 19.3(a)) [20]. For each sector's position, the following local parameters are calculated: the average value

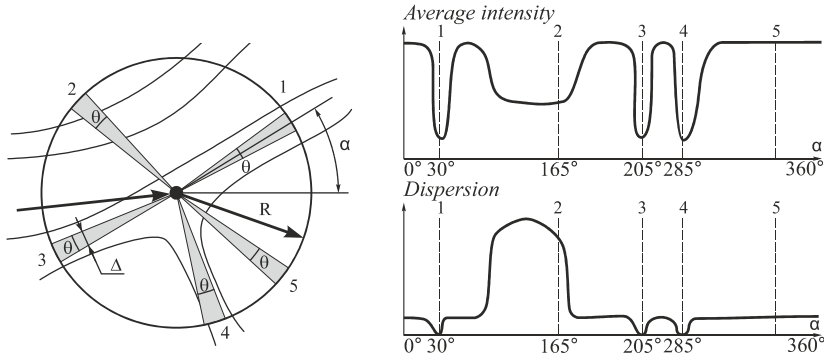


Figure 19.3. A circular region with differently oriented sector.

and variance of the brightness function $f(x, y)$. By analyzing these parameters as a function of angle, it is possible to detect the vessel at a given point, estimating its width and direction. A modified LFT is defined by the following formula:

$$F(x_0, y_0, \alpha, \theta, r) = \frac{1}{S_\theta} \int_{\alpha-\theta/2}^{\alpha+\theta/2} \int_0^r f(x_0 + t\cos\varphi, y_0 + t\sin\varphi) dt d\varphi,$$

$$D(x_0, y_0, \alpha, \theta, r) = \frac{1}{S_\theta} \int_{\alpha-\theta/2}^{\alpha+\theta/2} \int_0^r [f(x_0 + t\cos\varphi, y_0 + t\sin\varphi) - F(x_0, y_0, \alpha, \theta, r)]^2 dt d\varphi, \quad (19.1)$$

where (x_0, y_0) is a point of measurement, α is a polar angle, θ is a solid angle of the sector, r is the radius, and $S_\theta = \theta R^2/2$ is the sector's area.

Estimating the local direction of a vessel. The MLFT-aided method enables a local vessel direction to be evaluated. When compared with the LFT-aided approach, the MLFT method enables the radial profile with more pronounced minima to be obtained, with the local minima corresponding to bifurcation directions (see Figure 19.4). Thus, for a vessel direction to be identified, an optimization problem of searching for minima needs to be solved for each angle-dependent radial profile for a specific radius.

The algorithm operates by analyzing a list of directions, which represents a vector for each radius, with its length being equal to the number of directions and its magnitude taking a unit value if a bifurcation is detected and being zero otherwise. The list of directions

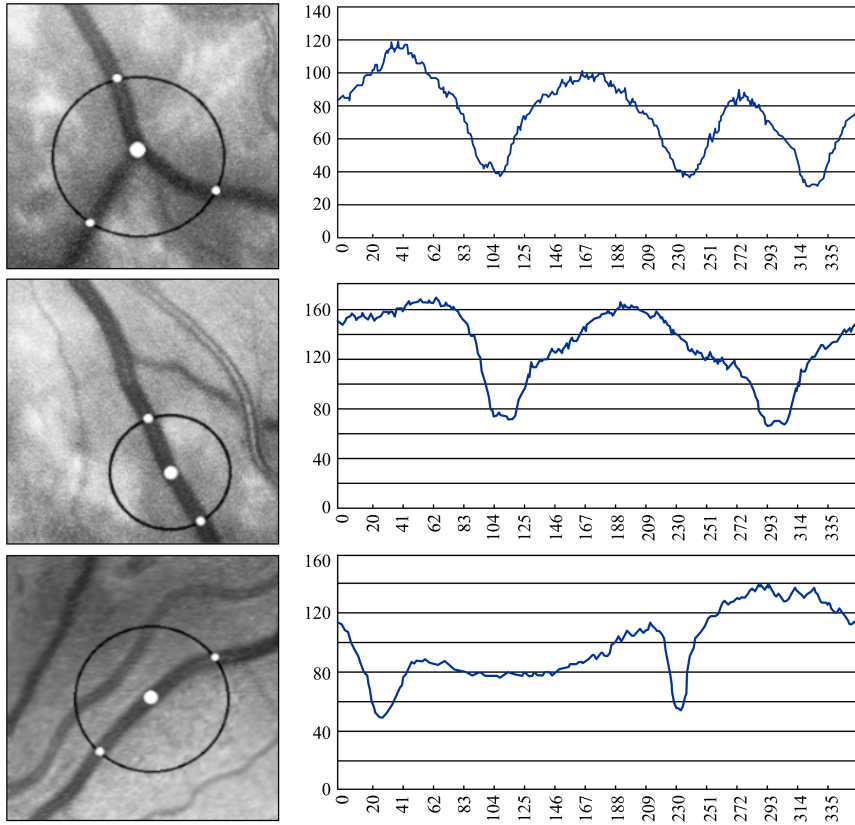


Figure 19.4. Vessel fragments with corresponding radial profiles.

may contain regions of constant values equal to unity. This is the indication of detecting a bifurcation with larger-than-pixel width. In this case, the unit value is taken just at the central region's pixel.

Estimating the vessel width. For estimation of vessels diameters is used LFT based on formula (19.1), where $\theta = 0$. Assume that the vessel lies horizontally — i.e. it is parallel to an axis OX. Having expressed (19.1) analytically at $\theta = 0$, we shall receive (19.2), where d_1 — part of the vessel diameter above OX, d_2 — part of the vessel diameter below OX, f_0 — background intensity, and f_1— vessel intensity.

Consider the system (19.3), there are three unknown variables d, f_0, f_1, solve them relative to d we find vessel diameter. In practice, vessels have random direction, for this purpose, the algorithm of

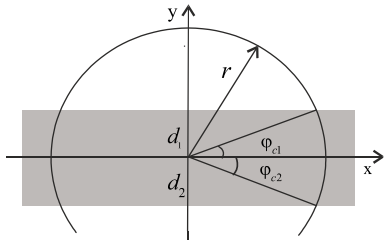


Figure 19.5. Horizontal vessel section.

vessel diameter measurement gets direction as a parameter — angle α, which was found by the algorithm discussed above. In this case system (19.4) turns on the angle α, and the final vessel diameter formula is (19.5) (see Figure 19.5).

$$F(\varphi,r)=\begin{cases}(r-d_1/\sin\varphi)f_0+f_1d_1/\sin\varphi, & \varphi\in(\varphi_{c1},\pi-\varphi_{c1})\\ (r-d_2/\sin(\varphi-\pi))f_0+f_1d_2/\sin(\varphi-\pi), & \\ \quad \varphi\in(\varphi_{c2},\pi-\varphi_{c2}) & \\ rf_1, \varphi\in(0,\varphi_{c1})\cup(\pi-\varphi_{c1},\pi+\varphi_{c2})\cup(2\pi-\varphi_{c2},2\pi)\end{cases}, \tag{19.2}$$

$$\begin{cases}F(0,r)=2rf_1\\ F(\pi/2,r)=(2r-d)f_0+df_1\\ F(\varphi,r)=(2r-d/\sin\varphi)f_0+f_1d/\sin\varphi, \quad \varphi\in(\varphi_{c\max},\pi-\varphi_{c\max})\end{cases},$$
$$\varphi_{c\max}=\max(\varphi_{c1},\varphi_{c2}) \tag{19.3}$$

Thus, having measured value F for three values φ, we can find diameter:

$$d=2r\frac{F(\varphi+\alpha,r)-F(\pi/2+\alpha,r)}{\frac{F(0+\alpha,r)-F(\pi/2+\alpha,r)}{\sin\varphi}-F(0+\alpha,r)+F(\varphi+\alpha,r)},$$
$$\varphi_{c1}=\arcsin\frac{d_1}{r}, \quad \varphi_{c2}=\arcsin\frac{d_2}{r}. \tag{19.4}$$

If equation (19.4) is employed in a straightforward manner, the components of the ray and fan transforms $F(\alpha,r)$ taken for a heavily noised vascular image can be calculated with an error, resulting in an incorrectly evaluated vessel's diameter. To avoid this in this work, the values of the LFT are averaged over points located uniformly on the perpendicular line of integration.

19.4 Methods to Evaluate the Three-Dimensional Features of Blood Vessels

The analysis of the local diameter of a blood vessel is a prevailing method at present. This approach can't give sufficient information to understand the development of many diseases. Other statistical parameters such as diameter variations of the vessel (venous beading, arterial idiospasm), its curvature and sinuosity [4] are important clinical indications, which help to estimate the current stage of the pathology and determine the probable course of the disease. It is also important to develop new effective integral indications that would allow us to estimate the retinal vascular system condition using images of the eye ground and coronary heart vessels. This kind of indication can be used as a basis for designing automated diagnostic systems, which would make it possible to standardize the diagnostic process and shorten the examination time. A single approach is offered for analysis of the two classes of images: microcirculatory system of the eye and cordial vascular system.

19.4.1 *The mathematical model of a blood vessel segment*

A mathematical model to represent blood vessels is offered. The model uses the division of geometric characteristics of a vessel into local and integral (global) parameters. The local parameters are computed directly in the course of blood vessel tracing, and integral parameters are used to form classification features. To form local parameters, let us introduce such notions as characteristics of the vessel path and configurations of its boundaries, which we call the walls of the vessel:

$$\begin{cases} x_1^b = x_1^b(t), & y_1^b = y_1^b(t) \\ x_2^b = x_2^b(t), & y_2^b = y_2^b(t) \end{cases}, \quad \begin{cases} x_1^b(t) = x(t) - r(t)\sin\varphi, \\ y_1^b(t) = y(t) + r(t)\cos\varphi. \end{cases}$$

We define the path parametrically by setting a central line and function of the vessel thickness determined as the distance from the path to the boundary of the branch perpendicular to the route:

$$\begin{cases} \gamma(t) = (x(t),\ y(t)), \left\| \frac{\partial\gamma}{\partial t} \right\| \neq 0 \\ r = r(t), 0 \leq t \leq L_v, \end{cases}$$

where t is a natural parameter, the distance measured along the path from its origin, and L_v is the length of the path.

The central line of the vessel is a line formed by the middle points of the vessel's transverse sections. This line can be represented as a continuous regular curve with natural parameterization. Then it is possible to uniquely determine such local parameters of the vessel as the direction of the path at each point $\varphi(t), tg\,\varphi(t) = -(dx(t)/dt)/(dy(t)/dt)$, local height $f(t)$ (it is determined as the distance from the current point of the path to its projection onto line segment L going from the start point to the end point of the path), and curvature of the vessel $k(t)$, $k(t) = d^2\gamma(t)/dt^2$. These parameters are evaluated by the image in the course of vessel tracking and permit us to define such geometric characteristics of the vessel as the variation of the vessel thickness $S = \sqrt{\overline{r^2} - \bar{r}^2}/\bar{r}$, mean curvature $K_r = \frac{1}{L_v}\int_0^{L_v} \left(\frac{d^2 y_t}{dx_t^2}\right) \Big/ \left[1 + \left(\frac{dy_t}{dx_t}\right)^2\right]^{\frac{3}{2}} dt$, straightness $Pr = L_v/L$, and configurations of the vessel path and walls. For describing the configurations of the vessel path and walls let us introduce the assumption about the harmonic behavior of the functions of direction and thickness: $\hat{r}(t) = A_0 \cos(\omega_0 t + \alpha_2)$, $\hat{\varphi}(t) = A_1 \cos(\omega_1 t + \alpha_1)$. We will consider only the segments of a vessel that correspond to the fundamental harmonic with the frequency proportional to the number of extremums of these functions according to the original definition of tortuosity in medicine. It means that we ignore insignificant changes that usually happen at the end of the vessel. So we can define the following geometric characteristics of vessels that are convenient in visual diagnostics: the degree of bending of the vessel wall A_0, tortuosity of the vessel wall ω_0, the degree of bending of the vessel path A_1, and tortuosity of the vessel path ω_1. They are defined as amplitude (A_0, A_1 are thickness and path amplitudes) and frequency (ω_0, ω_1 are thickness and path frequency) characteristics of the thickness function and direction function. To examine the three-dimensional vessel structure, let us introduce a spatial model of vessels (see Figure 19.6) determined by the following parametric

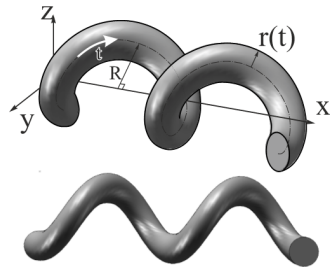


Figure 19.6. The three-dimensional model of a vessel.

equation of the vessel path:

$$\begin{cases} x = t, \\ y = A_1 \sin \omega_1 t \sin \omega_r t, \\ z = A_1 \sin \omega_1 t \cos \omega_r t, \\ r = \bar{r} + A_0 \sin \omega_0 t, \end{cases}$$

where A_1 is the amplitude of path variations, A_0 is the amplitude of thickness variations, ω_1 is the frequency of path variations, ω_0 is the frequency of thickness variations, and ω_r is the frequency of path rotation. Diagnosing pathology usually involves inspecting the angiogram and, therefore, is rather a subjective process depending on the angle of view of the recorded X-ray. The purpose of the spatial model of the vascular system is to give visual geometric and topologic information allowing higher accuracy of measurements and numerical estimates of the blood vessel geometry.

Papers [20, 21] present vessel tracing algorithms and estimations of vessel global features based on the two-dimensional model of vessels. The local parameters derived from the tracing of a branch in both two- and three-dimensional cases are the diameter and direction of the branch at each point. The global features (the mean diameter, straightness, venous beading, amplitude and frequency of thickness variations, thickness tortuosity, amplitude, and frequency of path bends) are computed using local parameters. Articles [20, 21] define

these features for two-dimensional structures and describe the difference for three-dimensional models. Attention is also paid to such a specific feature of a three-dimensional image as "frequency of rotation", which characterizes the departure of the path from the plane.

19.4.2 *Methods to evaluate the features*

All the features considered in the paper are geometrical and computed by using conventional methods of mathematical analysis and appropriate numerical methods for finding distances and derivatives. The discretization errors, quantization noise, and input device noise result in the processed data also holding many errors and, therefore, the numerical methods giving rather inaccurate results. Using one of the most popular features describing the varying shape of a vessel, the curvature as an example it is possible to show that the relative error of the feature grows indefinitely at points of vessel bending, the number of such points growing as the discretization interval decreases. Evaluating the second- and first-order derivatives with numerical methods and assuming for simplicity that vertical and horizontal discretization intervals are equal, we get the following definition of the curvature:

$$K_r(x) = \frac{y_1 + y_3 - 2y_2}{h^2} \Bigg/ \left[1 + \left(\frac{y_3 - y_1}{2h}\right)^2\right]^{\frac{3}{2}},$$

where $y_2 = y(x), y_1 = y(x-h), y_3 = y(x+h)$, $\Delta y = h/2$ is the greatest error in measuring the coordinates.

With the derivatives being approximated by finite differences, the computation of the feature gives the relative error [15]:

$$\begin{aligned}\delta K_r &= \sum_{i=1}^{3} \left|\frac{\partial}{\partial y_i} \ln K_r(y_1, y_2, y_3)\right| \Delta y_i \\ &= \frac{1}{K_r}\frac{4}{h^2}\frac{1}{\left[1 + \left(\frac{y_3 - y_1}{2h}\right)^2\right]^{\frac{3}{2}}} = \frac{4\Delta y}{y_1 + y_3 - 2y_2} = \frac{2}{hy''}.\end{aligned}$$

This allows the conclusion that the relative error of the feature grows indefinitely at the vessel bending points. Therefore, inaccurate input data makes the use of conventional definitions of mathematical

analysis and numerical methods for figuring out the features unreasonable. For this reason, it is necessary to develop noise-resistant algorithms for evaluating the features. It is also necessary to use global characteristics (which are derived from a large body of input data using the whole diagnostic image) rather than local geometric parameters (local curvature, local direction, etc.) to make the evaluation of diagnostic features more accurate.

In both two- and three-dimensional cases, a small enough segment of the vessel can be considered as a portion of a cylinder. The projection of a cylinder onto a plane not perpendicular to its axis is a figure whose transverse size is equal to the diameter of the cylinder (except the end portions). That is why all thickness features of a branch (the mean diameter, venous beading, thickness amplitude, thickness frequency, (wall) thickness tortuosity, and straightness) look the same for both two- and hree-dimensional tree structures (see Table 19.1).

Let us define the vessel features for three-dimensional vascular systems that supplement the vessel features introduced for two-dimensional systems. Let us first introduce the path features for a vessel segment (the amplitude and frequency of path variations) for a 3D vascular tree structure and also the feature describing the off-plane departure of the path, the frequency rotation.

The amplitude of path variations (for a 3D tree). To estimate the path variation amplitude, let us introduce the function of distance of a path point from axis Ox (see Figure 19.1): $R(x) = \sqrt{y^2(x) + z^2(x)}$. From the parametric equation of the path we get $R(x) = A|\sin \omega x|$. It follows from this expression that the mean value is $\bar{R}(x) = 2A/\pi$ and the path variation amplitude is $A_1 = \pi \overline{R}/2$.

The rotation frequency of the path (for a 3D tree). To calculate the frequency of oscillation of the path $\omega 1$ in the threedimensional case, we will use the Fourier transform. The function R(x) is periodic, but not sinusoidal, therefore, to improve accuracy use the following approach: there is the Fourier transform from the centered function where $R^2(x) = (1/2) \cdot A_1^2(1 - \cos 2\omega_1 x)$. Then, we obtain the estimate of the frequency of the track:

$$\omega_1 = \frac{1}{2} \arg\max_m \left| \sum_{n=0}^{N-1} \overline{R_n^2} e^{-\frac{2\pi i}{N} mn} \right|, \text{ where } \overline{R_n^2} = -(1/2) \cdot A_1^2 \cos 2\omega_1 t_n.$$

The Path Rotation Frequency. Theoretically, the rotation frequency of the path can be evaluated in the same way as the frequency of the

Table 19.1. Vessel global features.

Mean diameter	$\bar{D} = 2\bar{r} = \frac{2}{N}\sum_{n=1}^{N} r(t_n)$
Venous beading	$S = \sqrt{\overline{r^2} - \bar{r}^2}/\bar{r}$
Path straightness	$Pr = L_v/L = \frac{\sum_{n-1}^{N-1}\sqrt{(x_n - x_{n+1})^2 + (y_n - y_{n+1})^2}}{\sqrt{(x_1 - x_N)^2 + (y_1 - y_N)^2}}$
Thickness variation amplitude	$A_0 = \sqrt{2\overline{r^2} - 2\bar{r}^2}$
Thickness variation frequency	$\omega_0 = \frac{2\pi}{N}\arg\max_{1<m<N} R(m)$, $R(m) = \left\|\sum_{n=0}^{N-1} r(t_n)\exp\left(-i\frac{2\pi nm}{N}\right)\right\|$
Thickness tortuosity	$I_0 = A_0\omega_0$
Path amplitude	$A_1 = 2\bar{f}\cdot E(k)\Big/\left(1 + \frac{\ln(I_1 + \sqrt{1+I_1^2})}{I_1\sqrt{1+I_1^2}}\right)$, $\bar{f} = N^{-1}\sum_{n-1}^{N}\|f(t_n)\|$, where $E(k)$ is elliptical integral of the second kind
Path tortuosity	$I_1$is determined from the equation $Pr = (2/\pi)\cdot\sqrt{1+I_1^2}\cdot E(k)$, $k = I_1/\sqrt{1+I_1^2}$
Path frequency (for a 2D tree)	$\omega_1 = I_1/A_1$
Path frequency (for a 3D tree)	$\omega_1 = \frac{1}{2}\arg\max_m \left\|\sum_{n=0}^{N-1}\overline{R_n^2}e^{-\frac{2\pi i}{N}mn}\right\|$, $\overline{R_n^2} = -\frac{1}{2}A_1^2\cos 2\omega_1 t_n$
Path rotation frequency	$\omega_r = \sqrt{\omega_1^2 - \frac{k^2(1 + A_1^2\omega_1^2)}{A_1^2}}$, $k = E^{-1}(\pi Pr/2\sqrt{1 + A_1^2\omega_1^2})$

path: $\omega_r = (1/2)\cdot\arg\max\Im(\sin\varphi(x))$, where $\varphi(x)$ is the polar angle of the projection of a path point on the yOz plane. However, in most cases, real blood vessels are described by a low-frequency function of the path, which can lead to a large error in the evaluation of the feature.

For this reason, we will use the following approach. Straightness Pr of a curve (see the parametric equation of the path above) can be found as $Pr = (2/\pi) \cdot \sqrt{1 + A_1^2\omega_1^2}E(k)$, where $k^2 = A_1^2(\omega_1^2 - \omega_r^2)/(1 + A_1^2\omega_1^2)$, $E(\cdot)$ is the full elliptical integral of the second kind. Clear that when $\omega_r = 0$, we obtain the formula for the flat case.

Using the obtained estimates Pr, A_1, ω_1, we get the estimation of the rotation frequency of the path: $\omega_r = \sqrt{\omega_1^2 - (k^2(1 + A_1^2\omega_1^2)/A_1^2)}$, where $k^2 = E^{-1} \cdot (\pi Pr/2\sqrt{1 + A_1^2\omega_1^2})$, $E^{-1}(\cdot)$ is the inverse function of the second-kind elliptical integral.

So the algorithm for evaluating the feature consists of three steps:

(1) determining k by solving the equation for k numerically:

$$E(k) = \pi Pr/2\sqrt{1 + A_1^2\omega_1^2}. \tag{19.5}$$

(2) making the estimation using the formula:

$$\omega_r = \sqrt{\omega_1^2 - (k^2(1 + A_1^2\omega_1^2)/A_1^2)}.$$

The approach has a drawback. The integral of the second kind is known to reach its maximum at $k = 0$: $E_{\max} = E(0) = \pi/2$. Then equation (19.5) can't be solved for k when the estimation of straightness takes a certain value:

$$Pr = (2/\pi) \cdot \sqrt{1 + A_1^2\omega_1^2}E_{\max} = \sqrt{1 + A_1^2\omega_1^2}. \tag{19.6}$$

In this event, we go to the third step.

(3) Checking the estimation of straightness. If the estimation is in excess of the value computed by formula (19.6), the rotation frequency of the path is evaluated by using the expression:

$$\omega_r = \arg\max_m \left| \sum_{n=0}^{N-1} \sin\varphi(t_n) e^{-\frac{2\pi i}{N} nm} \right|^2 .$$

The effect of additive and pulsed noise on the quality of diagnostic feature estimations was investigated. Model blood-vessel images were used to perform the clustering of the feature space with the effect of noise on the clustering efficiency being examined. The experimental results confirmed the functional stability of the methods. With the

noise-to-signal ratio being no greater than 0.2, the features can be estimated to an average precision of no greater than 5%. The experiments showed that pulsed noise has the largest effect on the efficiency of feature estimations. It is necessary to use a median filter with a window size of 5 to reduce the effect of pulsed noise and increase the estimation accuracy. The test clustering of features was done to show that it is quite possible to use the features for determining the condition of vascular systems. With three visually distinguished model structures of a blood vessel being used as clustering centers, the classification error was 8%.

19.5 Formation of Features for Improving the Quality of Medical Diagnosis Based on Discriminant Analysis Methods

For the formation of new features aimed at solving specific problems, well-established methods based on discriminant analysis algorithm [10]. In practice, the use of one such method using the bruteforce of the feature combinations is possible to increase the accuracy of data classification [22–24]. It should be noted that when using the discriminant analysis algorithm capability of separating features (information content) is not changed, and higher reliability is achieved mainly by reducing the dimensionality of feature space, and, as a consequence, increasing the information content of the individual features. But the main drawback of a bruteforce of the features is a complete enumeration of total computational complexity. The exponential growth of the computational complexity makes it difficult to use the method when dealing with a large number of features. Therefore relevant is the task of developing an algorithm that allows us to obtain results similar to bruteforce of the features, but for a smaller number of operations.

Another disadvantage of the feature combinations bruteforce is particular algorithm discriminant analyses are not considered to reduce the computational complexity. In the literature for the formation of new features recommended to select eigenvectors based on eigenvalues of values corresponding to these vectors [10], and sort the eigenvalues descending [27]. This is due to the fact that the value of

the separability criterion attribute depends on the value of their own number corresponding to this criterion.

It is also necessary to note the following changes in the value of separability criterion using discriminant analysis algorithm:

(1) Criteria separability feature space cannot be less than the maximum values of the constituting separability criteria feature space.
(2) Criteria of separability of the feature space may be greater than the sum of the individual criteria of separability of features forming this space.

Therefore, for forming features, taking into account the contribution of each feature in the feature space separating ability, a method was developed directed enumeration, an algorithm which takes into account features of discriminant analysis. directed search method consists of three steps. In the first stage of the initial features of this feature space are excluded, the values of its own which the separability criterion does not exceed a certain threshold value e_1.

In the second step are eliminated those features which under exclusion value separability criterion signs remaining space is reduced to a value not exceeding a predetermined threshold value e_2.

Algorithmically second step is as follows:

(1) Common group separability criterion for all features of the original space is calculated. Go to step 2.
(2) Intra-Group separability criterion $J_i, i = \overline{1..N}$ for feature space with alternate each trait except where N — the total number of signs, i — a serial number of the excluded attribute is calculated. Go to Step 3.
(3) Such indication is sought, whose contribution to the common group criteria minimal $k = i, J_m - J_i \to \min$. If this contribution is less $e_2(J_m - J_i) < e_2$, a sign is excluded, and return to step 1. If there are, then step is completed.

The first two stages operate in a space of initial features, the third stage operates in the space of the formed features.

In the third stage, new features are formed on the basis of values of eigenvectors corresponding to the eigenvalues greater than a predetermined threshold value e_3. In contrast to the recommendations that take the necessary number of features found in the literature [1],

this step allows to select only formed informative features, regardless of their number. The values thus obtained eigenvectors are used for the formation of new features. One vector corresponds to one new feature. Thus formed is the desired feature space.

19.5.1 *Investigation of the method of directed enumeration*

For this chapter, this method has used samples, which consist of 4 groups of images 1000 of images in each group. The test features space consists of 20 features described above.

The first experiment is to compare the transformation matrices of the space obtained by directed enumeration and bruteforce. The parameter values used for the experiment. Bruteforce carried out by the method described previously [22, 23]. Transformation matrix obtained as a result of an experiment-directed enumeration method, it was identical space transformation matrix obtained by bruteforce $e_1 = e_2 = e_3 = 0,001$.

The second experiment is to evaluate the computational complexity of this method of directed enumeration. For this sample were formed the number of traits from 2 to 10, from those described above. To each sample was conducted informative features forming space with the use of directed enumeration and bruteforce. The number of repetitions for each algorithm and each space equal to 100. The parameters of directed enumeration were chosen as follows: $e_1 = e_2 = e_3 = 0$. The resulting acceleration value was calculated

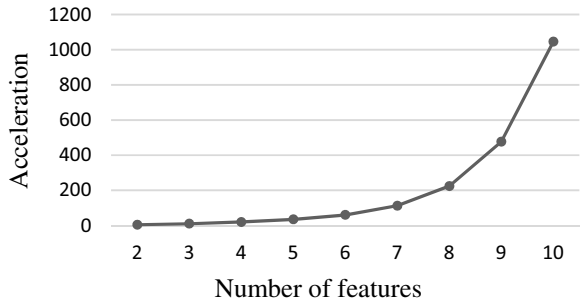


Figure 19.7. Graph of acceleration directed enumeration method for method of bruteforce for any number of attributes.

as the ratio of time spent by the bruteforce algorithm, to the time taken by the algorithm of directed enumeration. The resulting graph of acceleration is shown in Figure 19.7.

Thus, we can conclude that the directed enumeration method obtained no worse than the results obtained using the bruteforce method, but the first computational complexity is much smaller. This makes it possible to use the method to study the directed sorting feature spaces of large dimensions.

Acknowledgments

The work was carried out within the state assignment of NRC "Kurchatov institute".

References

[1] Aivazyan, S.A., Buchstaber, V.M., Enyukov, J.S. *et al.* (1989) Classification and dimensionality reduction. *Finance and Statistics*, 607 p.

[2] AlluraXper. electronic resource. http://www.healthcare.philips.com/ru_ru/products/interventional_xray/Product/interventional_cardiology/imaging_systems/intcardio_fd10.wpd.

[3] Astakhov, Y.S., Krasavina, M.I., and Grigoryeva N.N. (2009) Modern approaches to the treatment of a diabetic macular edema. *Ophthalmologic Sheets*, 59–69.

[4] Bribiesca, E. (2013) A measure of tortuosity based on chain coding. *Pattern Recognition*, 46(3), 716–724.

[5] Cheung, C.S., Butty, Z., Tehrani, N.N., and Lam, W.C. (2011) Computer-assisted image analysis of temporal retinal vessel width and tortuosity in retinopathy of prematurity for the assessment of disease severity and treatment outcome. *PubMed*, 15(4), 374–380.

[6] Elaheh, I., Malihe, J., and Hamid-Reza, P. (2015) Improvement of retinal blood vessel detection using morphological component analysis. *Computer Methods and Programs in Biomedicine*, 118(3), 263–279.

[7] Fathi, A. and Naghsh-Nilchi, A.R. (2013) Automatic wavelet-based retinal blood vessels segmentation and vessel diameter estimation. *Biomedical Signal Processing and Control*, 8(1), 71–80.

[8] Foracchia, M. (2001) Extraction and quantitative description of vessel features in hypertensive retinopathy fundus images. *Book Abstracts*

2nd International Workshop on Computer Assisted Fundus Image Analysis, p. 6.

[9] Fraz, M.M., Remagnino, P., Hoppe, A., Uyyanonvara, B., Rudnicka, A.R., Owen, C.G., and Barman, S.A. (2012) Blood vessel segmentation methodologies in retinal images. *Computer Methods and Programs Biomedicine*, 108(1), 407–433.

[10] Fukunaga, K. (1990) *Introduction to Statistical Patter Recognition* (Academic Press Professional, San Diego), 591 p.

[11] Gaidel, A.V., Kapishnikov, A.V., Pyshkina, Y.S., Kolsanov, A.V., and Khramov, A. G. (2018) Method of nephroscintigraphic dynamic images analysis. *Computer Optics*, 42(4), 688–694.

[12] Gaidel, A.V., Khramov, A.G., Kapishnikov, A.V., Kolsanov, A.V., and Pyshkina, Y.S. (2017) A method for digital renal scintigram analysis based on brightness and geometric features. *Computer Optics*, 41(1), 103–109.

[13] Gamma Multivox. Electronic resource. http://www.gammamed.ru/Multivox-Gamma-D1.html, http://www.multivox.ru/multivox2d.shtml.

[14] Haddouche, A., Adel, M., Rasigni, M., Conrath, J., and Bourennane, S. (2010) Detection of the foveal avascular zone on retinal angiograms using Markov random. *Digital Signal Processing*, 20(1), 149–154.

[15] Hart, W.E, Goldbaum, M., Côté, B., Kube, P., and Nelson, M.R. (1999) Measurement and classification of retinal vascular tortuosity. *International Journal of Medical Informatics*, 53(2–3), 239–252.

[16] Ilyasova, N.Y. (2014) Diagnostic computer complex for vascular fundus image analysis. *Biotehnosfera*, 3(33), 20–24 (in Russian).

[17] Ilyasova, N.Y. (2013) Methods for digital analysis of human vascular system. Literature review. *Computer Optics*, 37(4), 517–541.

[18] Ilyasova, N.Y., Kazanskiy, N.L., Korepanov, A.O., Kupriyanov, A.V., Ustinov, A. V., and Khramov, A.G. (2009) Computer technology for the spatial reconstruction of the coronary vessels structure from angiographic projections. *Computer Optics*, 33(3), 281–318 (in Russian).

[19] Ilyasova, N.Y., Korepanov, A.O., and Kupriyanov, A.V. (2009) A method of the wavelet transformation for estimation of geometrical parameters upon the diagnostic images. *Optical Memory & Neural Networks*, 18(4), 343–348.

[20] Ilyasova, N.Y., Kupriyanov, A.V., and Khramov, A.G. (2012) *Information Technologies of Image Analysis in Medical Diagnostics* (Radio I Svyazj), 424 p.

[21] Ilyasova, N.Y., Kupriyanov, A.V., Khramov, A.G., and Korepanov, A.O. (2009) *Computer Image Processing, Part II: Methods and*

algorithms: Appendix A2. Biomedical Images Processing (VDM Verlag, Saarbrücken), 584 p.

[22] Ilyasova, N.Y., Kupriyanov, A.V., and Paringer, R.A. (2014) Formation of features to improve medical diagnosis methods based on discriminant analysis. *Computer Optics*, 38(4), 851–856 (in Russian).

[23] Ilyasova, N.Y., Kupriyanov, A.V., and Paringer, R.A. (2015) The discriminant analysis application to refine the diagnostic features of blood vessels images. *Optical Memory and Neural Networks (Information Optics)*, 24(4), 309–313.

[24] Ilyasova, N.Y., Shirokanev, A.S., Kupriyanov, A.V., and Paringer, R.A. (2019) Technology of intellectual feature selection for a system of automatic formation of a coagulate plan on retina. *Computer Optics*, 43(2), 304–315.

[25] InterView Fusion. http://www.mediso.hu/products.php?fid=1,10,6&pid=67.

[26] Khorin, P.A., Ilyasova, N.Y., and Paringer, R.A. (2018) Informative feature selection based on the Zernike polynomial coefficients for various pathologies of the human eye cornea. *Computer Optics*, 42(1), 159–166.

[27] Kim, G.J., Myuller, C.W., Klekka, W.R., *et al.* (1989) Factor, discriminant analysis and cluster. *Finance and Statistics*, 215 p.

[28] Konstantinos, K., Aristides, I., Tsonos, C., and Assimakis, N. (2010) Automatic model-based tracing algorithm for vessel segmentation and diameter estimation. *Computer Methods and Programs in Biomedicine*, 100(2), 108–122.

[29] Kose, C. and Ikibas, C. (2011) A personal identification system using retinal vasculature in retinal fundus images. *Expert Systems with Applications*, 38(11), 13670–13681.

[30] Kupriyanov, A.V., Ilyasova, N.Y., and Ananin, M.A. (2007) Geometrical parameters estimation of the retina images for blood vessels pathology diagnostics. *Proceedings of 15th European Signal Processing Conference, EUSIPCO 2007,* pp. 1251–1254 (in Poland).

[31] Li, Q. (2010) Colour retinal image segmentation for computer-aided fundus diagnosis department of computing. The Hong Kong Polytechnic University, 126 p.

[32] Lupas, A., Tegolo, D., and Trucco, E. (2013) Accurate estimation of retinal vessel width using bagged decision trees and an extended multiresolution Hermite model. *Medical Image Analysis*, 17(8), 1164–1180.

[33] Myrian® XP-Vessel. http://www.eukon.it/site/download/Myrian_XP-Vessels.pdf.

[34] Niemeijer, M., van Ginneken, B., Cree, M.J., Mizutani, A., Quellec, G., Sanchez, C.I., Zhang, B., Hornero, R., Lamard, M., Muramatsu, C., Wu, X., Cazuguel, G., You, J., Mayo, A., Qin, L., Hatanaka, Y., Cochener, B., Roux, C., Karray, F., Garcia, M., Fujita, H., and Abramoff, M.D. (2009) Retinopathy online challenge: automatic detection of microaneurysms in digital color fundus photographs. *IEEE Transactions on Medical Imaging*, 29, 185–195.

[35] Ortega, M., Barreira, N., Novo, J., Penedo, M.G., Pose-Reino, A., and Gymez-Ulla, F. (2010) Sirius: A web-based system for retinal image analysis. *International Journal of Medical Informatics*, 79, 722–732.

[36] Perez-Rovira, A., MacGillivray, T., Trucco, E., Chin, K.S., Zutis, K., Lupascu, C., Tegolo, D., Giachetti, A., Wilson, P.J., Doney, A., and Dhillon, B. (2011) VAMPIRE: Vessel assessment and measurement platform for images of the Retina. *2011 Annual International Conference of the IEEE* (Engineering in Medicine and Biology Society, EMBC), pp. 3391–3394.

[37] Ruchay, A.N., Dorofeev, K.A. and Kolpakov, V.I. (2018) Fusion of information from multiple Kinect sensors for 3D object reconstruction. *Computer Optics*, 42(5), 898–903.

[38] Shirokanev, A.S., Kirsh, D.V., Ilyasova, N.Y., and Kupriyanov, A.V. (2018) Investigation of algorithms for coagulate arrangement in fundus images. *Computer Optics*, 42(4), 712–721.

[39] Smelkina, N.A., Kolsanov, A.V., Chaplygin, S.S., Zelter, P.M., and Khramov, A.G. (2017) Pulmonary emphysema recognition by CT scan. *Computer Optics*, 41(5), 726–731.

[40] Smelkina, N.A., Kosarev, R.N., Nikonorov, A.V., Bairikov, I.M., Ryabov, K.N., Avdeev, A.V., and Kazanskiy, N.L. (2017) Reconstruction of anatomical structures using statistical shape modeling. *Computer Optics*, 41(6), 897–904.

[41] Stewart, C.V. RIVERS: Retinal Image Vessel Extraction and Registration System/ Stewart, C.V. and Roysam B. http://cgi-vision.cs.rpi.edu/cgi/RIVERS/in-dex.php.in.

[42] Tsai, C.L., Madore, B., Leotta, M.J., Sofka, M., Yang, G., Majerovics, A., Tanenbaum, H.L., Stewart, C.V., and Roysam, B. (2008) Automated retinal image analysis over the internet. *IEEE Transactions on Information Technology in Biomedicine*, 12, 480–487.

[43] Vasilchenko, V.A., Burkovskiy, V.L., and Danilov, A.D. (2019) Algorithmization of the process of recognition of states of living objects based on special x-ray images. *Computer Optics*, 43(2), 296–303.

[44] Zakharov, A.A., Barinov, A.E., Zhiznyakov, A.L., and Titov, V.S. (2018) Object detection in images with a structural descriptor based on graphs. *Computer Optics*, 42(2), 283–290.

© 2025 World Scientific Publishing Company
https://doi.org/10.1142/9789811267215_0020

Chapter 20

Joint Flow Unwrapping and 2D Flow Restoration in Cardiac Color Doppler Flow Mapping

Artem M. Yatchenko[*,‡], **Andrey S. Krylov**[*,§], **Valery A. Sandrikov**[†,‖], **and Tatiana Yu. Kulagina**[†,¶]

[*]*Laboratory of Mathematical Methods of Image Processing, Faculty of Computational Mathematics and Cybernetics, Lomonosov Moscow State University, Moscow, the Russian Federation*

[†]*Federal State Budgetary Scientific Institution "Petrovsky National Research Center of Surgery", Moscow, the Russian Federation*

[‡]*yatchenko@cs.msu.ru*

[§]*kryl@cs.msu.ru*

[‖]*sandrikov@mail.med.ru*

[¶]*takula@list.ru*

Iterative regularization method to unwrap captured data by color Doppler flow mapping taking into account flow continuity has been proposed. The regularization functional includes terms that control captured flow velocities, smoothness of the flow field, and flow continuity. The constraint of ventricle border impermeability is used. Full-series filtration has been done for 4 patients with healthy hearts, 8 patients with different diseases, and an artificially generated series. GE Vivid E9 was used for capturing 10 heart cycles for each patient. Test results for the left ventricle color Doppler flow mapping investigation and for the artificial flow data showed the effectiveness of the proposed algorithm.

Keywords: Ultrasound, intracardiac flow, color Doppler, flow mapping, phase unwrapping, iterative regularization method.

20.1 Introduction

Cardiac color Doppler flow mapping (CDFM) is a method for non-invasively imaging blood flow through the heart by displaying flow data on the echo-cardiographic image. This capability has generated great excitement about the use of the technique for identifying different forms of heart disease, as the color flow image adds spatial information to the Doppler data [16]. The colors displayed on the flow map image contain useful information. By convention, Doppler color flow systems assign a given color to the direction of flow; in most systems, red is flow toward, and blue is flow away from the transducer. Progressively increasing velocities are encoded in varying hues of either red or blue. The more dull the hue, the slower the velocity. The brighter the hue, the faster the relative velocity. The intraventricular blood flow characteristics can be considered as one of the most informative sources for the estimation of the potential and kinetic miocard energy.

When the velocity range is set to high, low-flow states might not be displayed [18]. Conversely, when the velocity scale is set to low flow aliasing occurs. This phenomenon is related to the fact that Doppler technique is based on measuring the phase shift of emitted pulsed signals. If the interval between pulses is small and the measured flow velocity is high, the phase shift can exceed 2π. However, the transducer can estimate the phase shift only in the interval $[-\pi, \pi]$. So the so-called phase wrapping (or "aliasing") occurs [16, 20].

The phase unwrapping problem is crucial moment in CDFM image processing. Formally, we have:

$$v_r(x, y) = \psi(x, y) + 2\pi k(x, y), \tag{20.1}$$

Where $\psi(x, y)$ is the true phase in (x, y), $v_r(x, y) \in [-\pi, \pi]$ is measured wrapped modulo-2π phase, $k(x, y)$ is an integer value function. There are two main classes of phase unwrapping techniques in the literature: explicit and approximate. Explicit methods seek integer values for function $k(x, y)$ in each image pixel [2, 4, 7, 8, 11, 22, 26]. These methods do not work well when some data is missed or determined incorrectly.

Approximate methods seek real-values function $k(x, y)$ [10,17,21, 27]. In most areas, the function values are close to integer but in some areas $k(x, y)$ can differ significantly from the integer values. The methods can suppress noise and are less sensitive to incorrectly detected data areas.

In recent years, methods have emerged that use variable pulse-repetition-frequency to remove aliasing in ultrafast color Doppler [19].

Another drawback of CDFM is that it provides only the flow velocity projection parallel to the direction of the ultrasound beam. Such a 1D approach constitutes a particular limitation of echocardiography given the 3D nature of intracardiac flows. Nevertheless, studies using either numerical simulation or in vitro experiments have suggested that potentially relevant information can be derived from the full spatial characterization of intraventricular fluid dynamics [1, 9, 12, 23, 25] and the flows in the large vessels [5].

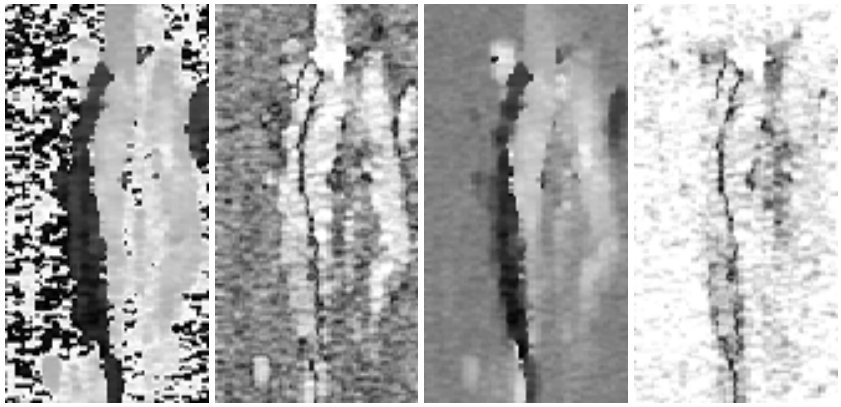
The method proposed in this chapter combines phase unwrapping and 2D flow restoration. 2D flow detection is close to [9] and phase unwrapping uses regularizing techniques based on [27]. But unlike 2D flow detection method by [9] continuity equation is used and normal components of velocities are included in the unwrapping regularization functional.

20.2 Unwrapping and Flow Restoration Method

20.2.1 *Data filtration*

Most of the common CDFM systems rely on estimating the rate of phase change of the signal returning from a given sample volume [6]. Scanning transducer estimates phases and amplitudes of the returned echoes. Several impulses (8–16) are formed for better stability. For the blood flow the unwanted clutter signal from solid tissues can exceed the signal from blood by 40–60 dB [3,13,14] due to the rejection of echoes from stationary or nearly stationary tissue. The use of the clutter rejection filters will color the system noise leading to a bias in the mean frequency estimate [15, 24]. As the result, a noise appears in the areas with low flow velocity.

To suppress this effect the method, described in [28] was used as the initial processing step. To restore original phase and amplitude



(a) (b) (c) (d)

Figure 20.1. Phase (a) and amplitude (b) of initial frame and phase (c) and amplitude (d) of restored frame.

of the signal, a mean bias value is detected first and then this value is used to restore the value of each pixel.

This preprocessing procedure suppresses the noise so the low velocities become visible and the amplitude becomes more reliable. The example of results is illustrated in Figure 20.1.

20.2.2 *Joined unwrapping and flow restoration*

We consider that the flux of mass is normal to the view plane and it can be neglected in the mass conservation equation. It enables us to reconstruct the second component of the blood velocity from echocardiographic color Doppler images [9].

Adopting a polar coordinate system centered at the head of the cone-beam ultrasound probe, the Doppler signal provides the radial component v_r of the blood velocity $\phi = [v_\varphi, v_r]$.

Let m be the measured signal. Taking into account (20.1) we are looking for a smooth solution $v_r(\phi)$, connected with the initial signal m by the following relationship

$$v_r(\phi) \approx m(\phi) + 2\pi k(\phi), \tag{20.2}$$

where $k(\phi)$ is a close to integer-valued function.

We define function $g(\varphi, r) = (g_\varphi(\varphi, r),\ g_r(\varphi, r))$.

$$g_\varphi(\varphi, r) = \frac{m(\varphi + \Delta\varphi, r) - m(\varphi, r) + 2\pi\, k_\varphi(\varphi, r)}{\Delta\, \varphi},$$

$$g_r(\varphi, r) = \frac{m(\varphi, r + \Delta r) - m(\varphi, r) + 2\pi k_r(\varphi, r)}{\Delta\, r},$$

where integer-valued functions $k_\varphi(\varphi, r)$ and $k_r(\varphi, r)$ limit upper parts of fractions to $[-\pi, \pi]$. Phase wrapping does not affect the derivatives of the signal, therefore partial derivatives of v_r should be close to g_φ and g_r, or

$$L_{g\varphi} = \frac{\partial}{\partial\varphi} v_r = g_\varphi, \quad L_{gr} = \frac{\partial}{\partial r} v_r = g_r. \tag{20.3}$$

Azimuthal velocities v_φ cannot be measured by CDFM since they are perpendicular to the ultrasound beams. They can be recovered at any specific instant of the cardiac cycle using the 2D continuity equation in polar coordinates

$$L_{\text{equ}}\phi = \frac{\partial}{\partial\varphi} v_\varphi(\phi) + \frac{r\partial}{\partial r} v_r(\phi) + v_r(\phi) = 0. \tag{20.4}$$

Assume that we already know some exact values of v_r in an area E:

$$v_r(\phi) = v_{r0}(\phi) \quad \text{for} \quad (\varphi,\, r) \in E \quad \text{or} \quad L_{v_r}\ \text{const}\phi = v_r(\phi) = v_{r0}(\phi). \tag{20.5}$$

The set E and the values v_{r0} are calculated iteratively. This process is described below in Section 20.2.4.

We also take into account that blood cannot pass through the ventricle, so the flow stream is parallel to the ventricle border, or

$$L_{\text{bound}}\phi = n_r(\phi) v_\varphi(\phi) + n_\varphi(\phi) v_r(\phi) = 0, \tag{20.6}$$

on the borders, where $(n_r; n_\varphi)$ is a normal of the boundary in the point ϕ. The border of the left ventricle is detected semi-automatically using border tracking algorithm.

The investigated field is blood flow, therefore it should be smooth:

$$L_{\text{smooth}}\phi = \frac{\partial^2}{\partial r^2} v_\varphi(\phi) = 0. \tag{20.7}$$

We do not include the derivatives of v_r in the equation (20.7), because the smoothness of the v_r in both directions is already considered in (20.3) and we do not include the partial derivative of v_φ in φ direction, because the smoothness of the v_φ in this direction follows from the equation (20.4).

We gather all listed above constraints (20.3), (20.4), (20.5), (20.6), and (20.7) and pose the problem of solution of (20.2) and v_φ restoration as the minimizing problem for the functional

$$\begin{aligned} M = {} & w_g \left\| L_g \phi - g \right\|_2^2 + w_{v_r\text{const}} \left\| L_{v_r\text{const}} \phi - v_{r0} \right\|_2^2 \\ & + w_{\text{equ}} \left\| L_{\text{equ}} \phi \right\|_2^2 + w_{\text{smooth}} \left\| L_{\text{smooth}} \phi \right\|_2^2 + w_{\text{bound}} \left\| L_{\text{bound}} \phi \right\|_2^2, \\ & \times M(\phi) \to \min, \end{aligned} \tag{20.8}$$

where w_g, $w_{v_r\text{const}}$, w_{equ}, w_{smooth}, w_{bound} are the regularizing parameters. These parameters reflect both physics of the signal process and the resulting solution constraints. We found the optimal values of method parameters: $w_g = 1$, $w_{v_r\text{const}} = 100$, $w_{\text{equ}} = 1$, $w_{\text{smooth}} = 0.1$, $w_{\text{bound}} = 10$ using test and practical data described later in the article.

20.2.3 *Minimization*

All operators L_{dv_r}, $L_{v_r\text{const}}$, L_{equ}, L_{smooth} and L_{bound} are linear, therefore minimization problem (20.8) can be solved as a solution of equation

$$\begin{aligned} Q\phi &= \rho, \\ Q &= w_g^2 L_g L_{dv_r}^T \phi + w_{v_r\text{const}}^2 L_{v_r\text{const}} L_{v_r\text{const}}^T \phi \\ &\quad + w_{\text{equ}}^2 L_{\text{equ}} L_{\text{equ}}^T \phi + w_{\text{smooth}}^2 L_{\text{smooth}} L_{\text{smooth}}^T \phi \\ &\quad + w_{\text{bound}}^2 L_{\text{bound}} L_{\text{bound}}^T \phi, \\ \rho &= w_g L_g^T \, g + w_{v_r\text{const}} L_{v_r\text{const}}^T v_{r0}. \end{aligned} \tag{20.9}$$

Equation (20.9) can be solved using conjugate gradient method.

20.2.4 *Iterative construction of the set E*

The set E contains pixels with already detected true phases $v_{r0}(\varphi, r)$. Initially, we do not know exact values of the velocities, so the set E is empty.

Further this set and values $v_{r0}(\varphi, r)$ are determined iteratively. Initially, E is empty and we solve the equation (9) $Q\phi = \rho$, $\phi_0(\varphi, r) = 0$ without incorporation of $L_{v_r \mathrm{const}}$.

Then we update the set E and values $v_{r0}(\varphi, r)$ as follows. We fix 30% of velocities v_r that are the modulo 2π closest to $m(\varphi, r)$.

$$E(\varphi, r) = \{(\varphi, r) \colon |m(\varphi, r) - \phi(\varphi, r) - 2\pi k(\varphi, r) \to \min\},$$

where integer valued function $k(\varphi, r)$ gives

$$|m(\varphi, r) - \phi(\varphi, r) - 2\pi k(\varphi, r)| < \pi, \quad v_{r0} = m(\varphi, r) + 2\pi k(\varphi, r).$$

At each new iteration equation $Q\phi = \rho$ is solved for the new calculated E and $v_{r0}(\varphi, r)$ and the values of E and $v_{r0}(\varphi, r)$ are refined. We stop the process after 10 iterations when 97% of velocities v_r are fixed.

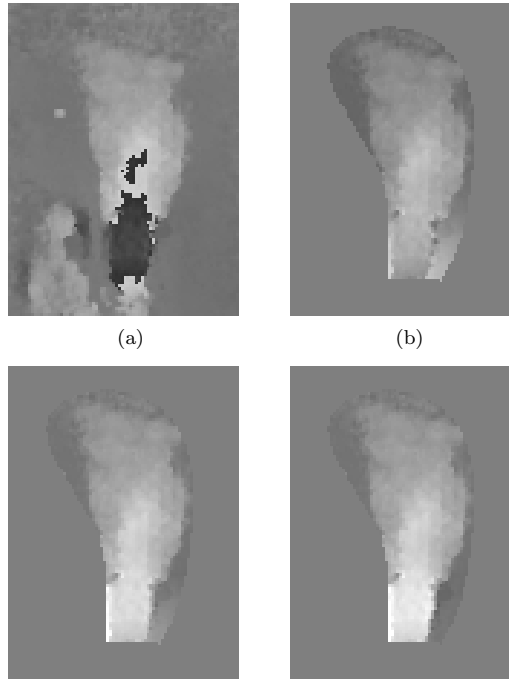
Results at different iteration steps are illustrated in Figure 20.2.

20.3 Test Data Generation

To compare the quality of joined phase unwrapping and flow restoration result, we have generated artificial ultrasound slices.

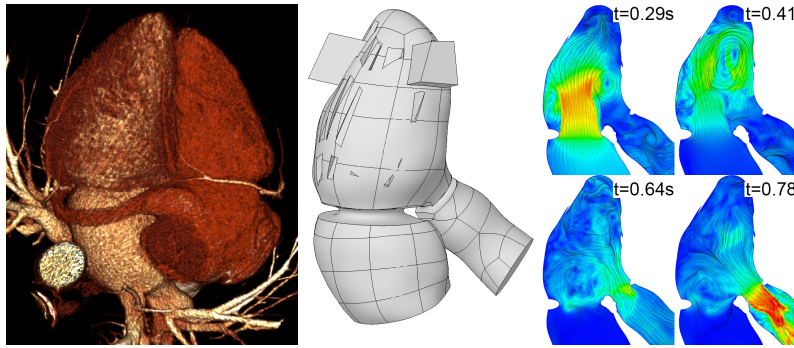
We used a real patient CT series to build a 3D heart model (see Figure 20.3(a)). The heart geometry was static for the flow calculations, therefore additional open areas were added in the apex of the ventricle to let the redundant blood exit and enter the chamber during the systole and diastole. Final geometry is illustrated in Figure 20.3(b). The details of the flow generation are described in [29]. Inflow and outflow with the sine-form shape 0.5 s duration and 1 m/s peak velocity were used. 7 total heart cycles (1 second each cycle) were simulated (see Figure 20.3(c)).

For the calculated flow streams a single 2D plane was fixed and virtual US scans of a convex probe were simulated. The scanning plane imitated the 3-chamber probe position (see Figure 20.4(a)). Frame rate was limited to 14.29 fps (total 100 frames per 7 cycles)



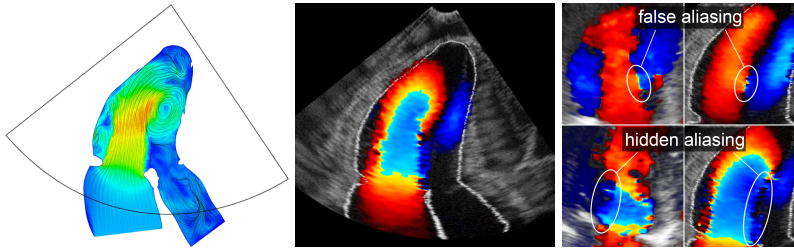
(a) (b) (c) (d)

Figure 20.2. Algorithm iterations example: (a) input wrapped data, (b) unwrapping result after the first iterations, (c) after 5 iterations, (d) after 10 iterations.



(a) (b) (c)

Figure 20.3. Model of heart: (a) anatomical volume rendering of CT slices, (b) heart model, (c) different time moments of cardiac cycle flow.



(a) (b) (c)

Figure 20.4. (a) selected plane and area of virtual US scan for $t = 0.33\,\mathrm{s}$, (b) artificial color Doppler flow mapping image, (c) false aliasing and hidden aliasing artifacts for real (left) and artificial (right) series.

that is a typical frame rate for real Doppler investigation. Real phase F, amplitude A and power P (these values are described in section 20.2.1) were constructed with the resolution 80 beams and 200 pixels per beam. Artificial color Doppler flow image was generated (Figure 20.4(b)) with the following added distortions corresponding to the real probe conditions: random shift of a measured point with maximum displacement 4 mm, region averaging, 7 dB white noise and velocity wrapped to $[-0.5, 0.5]\,\mathrm{m/s}$.

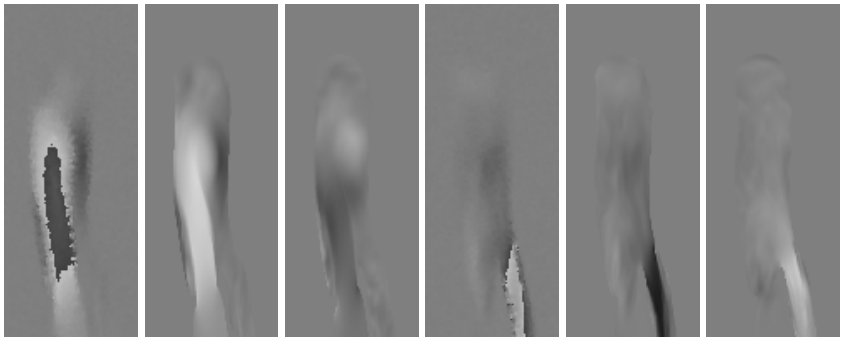
Generated artificial images have all drawbacks inherent to real ultrasonic Doppler scans: phase wrapping, noise, false aliasing, and hidden aliasing (see Figure 20.4(c)).

Undistorted images were also stored as a set of phase F fields with the same resolution as noisy fields. We use this set as reference data to validate the results of joined unwrapping and flow restoration method. The examples of distorted and undistorted frames of generated series are in Figure 20.5.

20.4 Experiments and Results

We used both test data and real series to validate the results of proposed method. To detect the quality of the method we use the following two parameters.

The first one is the mean deviation of the restored distorted phase: $E_{\mathrm{phase}} = |v_r(\phi) - \tilde{v}_r(\phi)|$, where $v_r(\phi)$ is an unwrapping result of the distorted artificial frame in the point ϕ and $\tilde{v}_r(\phi)$ is a real



(a) (b) (c) (d) (e) (f)

Figure 20.5. (a), (d) generated frames with distortion for the time moments $t = 0.25\,\mathrm{s}$ and $t = 0.75\,\mathrm{s}$ respectively, (b), (e) v_r values of undistorted frames for the same time moments, (c), (f) v_φ values of undistorted frames.

undistorted value at the same point. E_{phase} represents the quality of the phase unwrapping for the developed method.

The second tested parameter is a mean deviation of the absolute value of velocity:

$$E_{\mathrm{abs}} = |v(\phi) - \tilde{v}(\phi)|, \quad v(\phi) = \|(v_r(\phi), v_\varphi(\phi))\|_{L2}\,,$$
$$\tilde{v}(\phi) = \|(\tilde{v}_r(\phi), \tilde{v}_\varphi(\phi))\|_{L2}\,,$$

where $(v_r(\phi), v_\varphi(\phi))$ is a vector with components $v_r(\phi)$ and $v_\varphi(\phi)$, $v_r(\phi)$ and $\tilde{v}_r(\phi)$ are the same as in E_{phase}, $v_\varphi(\phi)$ is a reconstructed transversal velocity in the point ϕ and $\tilde{v}_\varphi(\phi)$ is a real undistorted transversal velocity in the same point. The absolute values of velocities are more informative for a diagnosis and E_{abs} represents the quality of detecting real absolute velocity in each point.

The parameters E_{phase} and E_{abs} were computed for three series. The first series is the initial distorted series with phase wrapping and with unknown transversal velocities. The second series is the result of regularizing unwrapping method [27]. The third series is the result of the proposed method. For the first two series transversal velocities are unknown, therefore $v_\varphi(\phi)$ values were set to 0 in these cases. The results of the E_{phase} and E_{abs} are illustrated in Figures 20.6 and 20.7.

Full-series filtration has been done for real series of 4 patients (A–D in Table 20.1) with healthy hearts and 8 patients (E–L in Table 20.1) with different diseases, age 21–73. GE Vivid E9 was used for capturing 10 heart cycles for each patient. Investigation was conducted with the following conditions: 3-chambered plane, variances

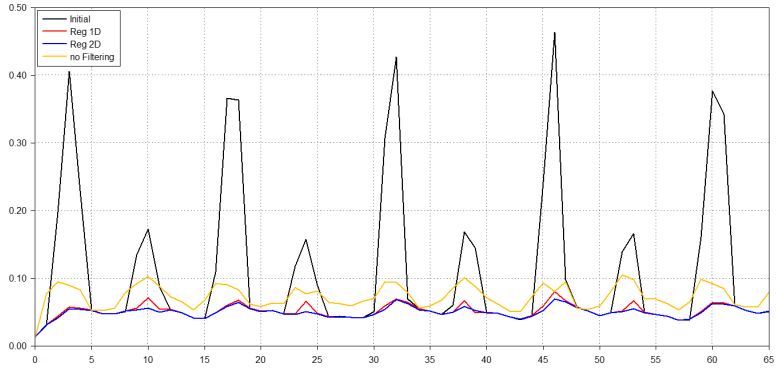


Figure 20.6. Graph of the E_{phase}. x axis represents the frame number, y axis represents the E_{phase} value for the frame. Initial — initial distorted series, Reg 1D — the result of [27] unwrapping, Reg 2D — the result of the proposed unwrapping method.

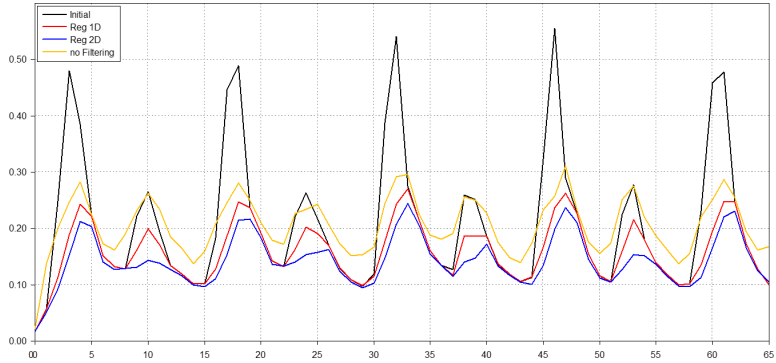


Figure 20.7. Graph of the E_{abs}. x axis represents the frame number, y axis represents the E_{abs} value for the frame. Initial — initial distorted series, Reg 1D — the result of [27] unwrapping, Reg 2D — the result of the proposed unwrapping method.

turned on, 11 frames per second, depth of scanning 18 mm., velocities interval [−45 m/s, 45 m/s] to [−72 m/s, 72 m/s]. The number of DICOM frames for one patient lied in the interval 66–137 with the average number 87. Each frame was processed approximately 8–15 s on 4-core 3.3 GHz CPU. Almost all of the results were classified by cardiologists as successful and all the results were more informative than the [27] method results.

Example of the result is in Figure 20.8.

Table 20.1. Comparison of the results for 12 patients. N — total number of frames in series, Reg 1D — number of improperly unwrapped frames with method [27], Reg G2D — number of improperly unwrapped frames with proposed method.

Patient	A	B	C	D	E	F	G	H	I	J	K	L
N	89	88	66	137	127	98	115	88	76	112	67	74
Reg 1D	1	7	5	12	7	0	10	2	1	8	3	4
Reg 2D	2	1	0	5	2	0	1	0	0	2	0	0

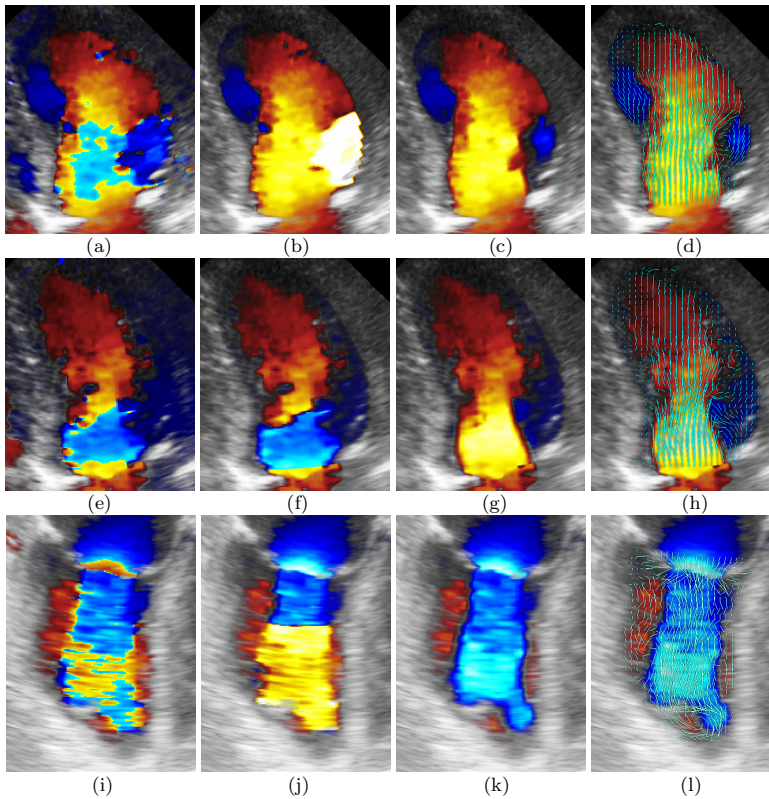


Figure 20.8. Results comparison on real data: (a), (e), (i) — the initial aliased frames, (b), (f), (j) — the unwrapping result of the 1D Regularizing method [27], (c), (g), (k) — the unwrapping result of the proposed method, (d), (h), (l) — the vector visualization of the blood flow.

20.5 Discussion

As can be seen from Figures 20.6 and 20.7, joined phase unwrapping and 2D flow restoration provide more accurate results than just phase unwrapping. Flow constraints make wrapped area detection more predictable and reduce unwrapping errors for the frames with the most intensive flow velocities.

2D flow restoration provides real flow velocities which are more diagnostically useful than CDFM values and can significantly vary from them for a flow directed to the transducer at an angle.

Complex phase restoration stage enhances the quality of the result for all frames and makes wrapped areas more clear.

20.6 Conclusions

A method for iterative phase unwrapping in 2D color Doppler flow mapping has been suggested. Analysis of the results was performed both with the artificial test data and with medical images of healthy patients and patients with different heart diseases. The studies were conducted at Federal State Budgetary Scientific Institution Petrovsky National Research Center of Surgery. The clinical studies showed that the proposed method better increases the CDFM image quality compared with the existing methods.

Acknowledgment

This work was supported by project No. FURG-2020-0007 of the Russian Ministry of Education and Science 2020–2022.

References

[1] Arigovindan, M., Suhling, M., Jansen, C., Hunziker, P., and Unser, M. (2007) Full motion and flow field recovery from echo doppler data. *IEEE Transactions on Medical Imaging*, 26, 31–45.

[2] Bioucas-Dias, J.M. and Valadao, G. (2007) Phase unwrapping via graph cuts. *IEEE Transactions on Image Processing*, 16, 698–709.

[3] Brands, P.J., Hoeks, A.P., Hofstra, L., and Reneman, R.S. (1995) A noninvasive method to estimate wall shear rate using ultrasound. *Ultrasound in Medicine and Biology*, 21, 171–185.
[4] Cusack, R., Huntley, J.M., and Goldrein, H.T. (1995) Improved noise-immune phase-unwrapping algorithm. *Applied Optics*, 34, 781–789.
[5] Ekroll, I.K., Avdal, J., Swillens, A., Torp, H., and Løvstakken, L. (2016) An extended least squares method for aliasing-resistant vector velocity estimation. *IEEE Transactions on Ultrasonics, Ferroelectrics, and Frequency Control*, 63(11), 1745–1757.
[6] Evans, D., Jensen, J., and Nielsen, M. (2011) Ultrasonic colour Doppler imaging. *Interface Focus*, 1, 490–502.
[7] Flynn, T.J. (1996) Consistent 2-D phase unwrapping and absolute values information. *Proceedings of IGARSS*, pp. 2057–2059.
[8] Flynn, T.J. (1997) Two-dimensional phase unwrapping with minimum weighted discontinuity. *Journal of the Optical Society of America A*, 14, 2692–2701.
[9] Garcia, D., del Alamo, J.C., Tanne, D., Yotti, R., Cortina, C., Bertrand, E., Antoranz, J.C., Perez-David, E., Rieu, R., Fernandez-Aviles, F., and Bermejo, J. (2010) Two-dimensional intraventricular flow mapping by digital processing conventional color-doppler echocardiography images. *IEEE Transactions on Medical Imaging*, 29, 1701–1713.
[10] Ghiglia, D.C. and Romero, L.A. (1996) Minimum Lp-norm two-dimensional phase unwrapping. *Journal of the Optical Society of America A*, 13, 1999–2013.
[11] Goldstein, R.M., Zebker, H.A., and Werner, C.L. (1988) Satellite radar interferometry: Two-dimensional phase unwrapping. *Radio Science*, 23, 713–720.
[12] Gomez, A., Simpson, J., Cheng, Y., Schaeffter, T., and Penney, G. (2011) 3d flow reconstruction from multiple registered echo Doppler views. *Biomedical Imaging: From Nano to Macro*, pp. 879–882.
[13] Heimdal, A. and Torp, H. (1997) Ultrasound Doppler measurements of low velocity blood flow: Limitations due to clutter signals from vibrating muscles. *IEEE Transactions on Ultrasonics, Ferroelectrics, and Frequency Control*, 44, 873–881.
[14] Jensen, J. (1993) Stationary echo canceling in velocity estimation by time-domain cross-correlation. *IEEE Transactions on Medical Imaging*, 12, 471–477.
[15] Kadi, A.P. and Loupas, T. (1995) On the performance of regression and step-initialized IIR clutter filters for color Doppler systems in

diagnostic medical ultrasound. *IEEE Transactions on Ultrasonics, Ferroelectrics, and Frequency Control*, 42, 927–937.

[16] Kisslo, J. and Adams, D.B. (1988) *Doppler Color Flow Imaging* (Churchill Livingstone, London).

[17] Marroquin, J.L. and Rivera, M. (1995) Quadratic regularization functionals for phase unwrapping. *Journal of the Optical Society of America A*, 12, 2393–2400.

[18] Maulik, D. (2005) Spectral Doppler: Basic principles and instrumentation. *Doppler Ultrasound in Obstetrics and Gynecology*, pp. 19–34.

[19] Posada, D., Porée, J., Pellissier, A., Chayer, B., Tournoux, F., Cloutier, G., and Garcia, D. (2016) Staggered multiple-PRF ultrafast color Doppler. *IEEE Transactions on Medical Imaging*, 35(6), 1510–1521.

[20] Pozniak, M.A., Zagzebski, J.A., and Scanlan, K.A. (1992) Spectral and color Doppler artifacts. *RadioGraphics*, 12, 35–44.

[21] Pritt, M.D. and Shipman, J.S. (1994) Least-squares two-dimensional phase unwrapping using FFTs. *IEEE Transactions on Geoscience and Remote Sensing*, 32, 706–708.

[22] Saad, A.A. and Shapiro, L.G. (2008) A new fringeline-tracking approach for color Doppler ultrasound imaging phase unwrapping. *Medical Imaging 2008: Ultrasonic Imaging and Signal Processing*, p. 6920.

[23] Tafti, P.D. and Unser, M. (2011) On regularized reconstruction of vector fields. *IEEE Transactions on Image Processing*, 20, 3163–3178.

[24] Tysoe, C. and Evans, D.H. (1995) Bias in mean frequency estimation of Doppler signals due to wall clutter filters. *Ultrasound in Medicine and Biology*, 21, 671–677.

[25] Wang, Y., Georgescua, B., Datta, S., Shizhen, L., Vannan, M.A., and Comaniciu, D. (2011) Automatic cardiac flow quantification on 3d volume color Doppler data. *Biomedical Imaging: From Nano to Macro*, pp. 1688–1691.

[26] Yatchenko, A.M., Krylov, A.S., Gavrilov, A.V., and Arkhipov, I.V. (2011) Graph-cut based antialiasing for Doppler ultrasound color flow medical imaging. *International Conference Visual Communications and Image Processing*.

[27] Yatchenko, A. and Krylov, A. (2013) Iterative phase unwrapping in color Doppler flow mapping. Lecture Notes in Computer Science, pp. 330–338.

[28] Yatchenko, A., Krylov, A., Gavrilov, A., Sandrikov, V., and Kulagina, T. (2014) Image preprocessing for color Doppler flow antialiasing using power and complex phase data. *International Conference on Signal Processing*, pp. 1072–1076.

[29] Yatchenko, A. and Krylov, A. (2015) Cross-frame ultrasonic color Doppler flow heart image unwrapping. Lecture Notes in Computer Science, pp. 265–272.

© 2025 World Scientific Publishing Company
https://doi.org/10.1142/9789811267215_0021

Chapter 21

Computer Analysis of Single Point Radiator Fluorescent Images

Aleksey A. Rozhentsov* **and Aleksey A. Baev†**

Volga State Technological University, Yoshkar-Ola, the Russian Federation

**RozhencovAA@volgatech.net*

†*kruzenshteyn@yandex.ru*

In this chapter, the main tasks of the analysis of nanoscopic images and approaches to their solution, providing the possibility of data processing in real time, are considered. Original algorithms for the detection and resolution of images of single-point irradiators, including those with a subpixel resolution, are proposed. The task of tracking and estimating the parameters of tracks of fluorescent molecules in nanoscopic images is solved. A technique for automating the estimation of the parameters of images of a "double helix" with the purpose of obtaining three-dimensional descriptions of nanoscopic objects is considered. Examples of processing nanoscopic images are given, and recommendations for ensuring the required speed of the computer system are given.

Keywords: Nanoscopic image, nanoscale objects, fluorescent microscopy, ultrafine resolution, spectroscopy of single molecules, stochastic optical reconstruction microscopy, photo activation localization microscopy, point spread function, stimulated emission depletion, instrumental response function, Levenberg–Marquardt algorithm, deconvolution, Hough transform.

21.1 Introduction

The new techniques of researching the nanoscale objects is of great importance in modern science. Due to the restrictions related to the existence of the diffraction limit, the possibilities of usual optical microscopy are practically exhausted and it is necessary to develop new methods to form and process the imaging of objects at nanoscale. In this case, nanoscopy is considered to be a promising technique which allows us to synthesize the images with a resolution higher than that of the diffraction limit on the base of the analysis of microscopic images having embedded fluorescent tags [1]. It is impossible to obtain such results without the application of new approaches to the digital processing of images which are based on the solution of some tasks including statistical analysis of background features, instrumental response function form, etc. Those are necessary to provide the possibilities for optimum solutions to the problems of filtering and detection of point and small-size tags, their resolution, and parameter estimation. Particular attention requires the solution of the processing of nanoscale object imaging on a real-time basis. This is due to the fact that there is a high rate of big data input and high intensity of used nowadays algorithms, necessity to control the procedure of an experiment in real time or near-real time mode. In this case, there is a need to develop high-performance calculating algorithms which would provide the required quality of nanoscale image processing and opportunity to parallelize the performed operations.

The aim of this chapter is to develop approaches for nanoscale image processing based on the techniques of appropriate signal processing which would provide the possibility to analyze them in real or near real-time mode.

21.2 Fluorescent Microscopy of Ultrafine Resolution

21.2.1 *The concept of fluorescent microscopy of ultrafine resolution*

Fluorescent microscopy of ultrafine resolution is one of the promising scientific issues nowadays. William Moerner (Stanford University, USA), Eric Betzig (Howard Hughes Medical Institute, USA), and

Stefan Hell (Max Planck Institute, Heidelberg, Germany) received the Nobel Prize in Chemistry in 2014 for their achievements in this field. Optical microscopy is one of the oldest experimental techniques which is still of great importance in many branches of science. Ever since the 1873 work of Abbe, it has become clear that the special resolution of a far-field scanning microscope based on the focusing optics has its limit which is defined by diffraction in the visible spectrum within $\sim 200 - 300$ nm.

So far, there have been developed a lot of techniques which can restore the sub-diffraction structure of a material including electronic, atom-force, near-field scanning, and x-ray microscopy. However, each of these techniques has critical disadvantages which make their common use in nanodiagnostics of structure and dynamics of the specimens difficult. It is especially problematic in cases which concern the efficiency of the research, simplicity of measuring procedure, need for structure diagnostics and object dynamics, as well as its surface and volume. The development of laser fluorescent microscopy techniques has allowed researchers to overcome the classical diffraction limit in optical microscopy.

21.2.1.1 *Single molecules*

The first group of optical nanoscopy techniques is based on the detection of single fluorescent molecules. To implement this idea, it is necessary to achieve separate (selective) detection of separate molecules, i.e. move from simultaneous registration of a huge number of molecule fluorescence to their minimum concentrations and further on to single molecules.

Persistent work in the field of luminescence of solutions was carried out in the laboratory of luminescence of Physical Institute of the RAS under the guidance of S.I. Vavilov. In 1952, researchers of the Department of Theoretical Physics in Moscow State Pedagogical Institute under the guidance of E.V. Shpolsky for the first time witnessed the effect of the appearance of quasi-line spectra of complex organic compounds in specially matched solvents at low temperatures. At cryogenic temperatures, the spectra of embedded molecules firmly fixed in a transparent matrix become very narrow. At this moment, it is possible to excite and make separate clusters of molecules (instead of all molecules) light up. In 1972, professor

Personov R.I., the disciple of Shpolsky, and his colleagues (Institute of Spectroscopy of USSR AS) succeeded in selective laser luminous excitation of organic molecules. Then, in 1974, Personov and his colleagues from the Institute of Physics, Tartu, ventured into the experiments with "burning spectral dips", when with the help of a laser a part of molecules "was transformed into non-radiative state" and did not radiate the light. These works formed the basis of the selective laser spectroscopy of molecules the greatest achievement of which has become spectroscopy of single molecules [19].

In 1989, Moerner and Kador showed the possibility of detecting the absorption spectrum of single molecules [18], and few months later, Orrit and Bernard in their experiment showed demonstratively the possibility of detecting the fluorescence of single molecules in a hard mould at cryogenic temperatures [21]. Since the first experiments 25 years ago, this field has been developing rapidly and has become one of the most up-to-date in physical chemistry, biophysics, and related branches of science. The growing interest in the spectroscopy of single molecules is due to a number of causes:

1. Physical and chemical properties of a substance can be studied at the lowest possible level — at the level of single molecules.
2. Optical spectra and spectral dynamics of single molecules embedded into the environment under investigation (liquid, solid) as nanoprobes contain unique information both on the radiators and parameters of their nearest environment.
3. The registration of the luminescent images of single point radiators in the circuit of the scanning confocal microscope or classical luminescent microscope with a 2D detector allows us to construct the coordinates of single molecules with sub-diffraction (nanometer) accuracy.

21.2.1.2 *Microscopy with single molecules image reconstruction*

Photons of luminescence of single molecules form an image of a point light source on a 2D detector. The analysis of a single molecule image with due regard to a specific form of the instrumental response function of the point source allows us to define the location (coordinates) of a single molecule with an error which is limited by the positional

integrity, the "signal-noise" ratio of the registered radiation and the size of the point radiator. The error can significantly exceed the diffraction limit.

At the same time, the diagnostic tasks of the extended objects, as a rule, require the detection of a significant number of single molecules and their high special density.

In this case, simultaneous luminescence of a thousand molecules does not allow us to define the images of single point sources, i.e. the identification and definition of the precise location of single molecules is impossible. It is possible to overcome this difficulty if different single molecules radiate at different times or if we differentiate single molecules which are inside the diffraction limit according to some extra feature or property [9].

There are several techniques which can help implement this idea. The most widespread at the moment is the technique which is based on the phenomenon of fluorescence glitter of a single quantum object. Accident interruption of single molecule fluorescence occurs at different moments of time. Thus, having a relatively low concentration of guest molecules at each moment of time inside each diffractionally limited volume of the specimen there is only one fluorescent molecule. It helps to fix its image and register its coordinates. This idea was implemented in a technique called photo activation localization microscopy (PALM) by a team of scientists under the guidance of Betzig [4]. It should be noted that there has been developed the technique called stochastic optical reconstruction microscopy (STORM) [25] which is based on the same principles as the previous one. Over the last few years, original methods have been developed for single molecule visualization which help us to define the coordinates of the point light source [20, 22].

21.2.1.3 *Stimulated emission depletion microscopy*

The second group of optical nanoscopy techniques was being developed along with the development of single-molecule techniques in the mid-2000s. This new group of methods was based on the manipulation of the parameters of the exciting radiation and nonlinear response of the environment under investigation. A specific set of the parameters of the exciting laser radiation helps us to significantly reduce the dimensions of the point spread function. For instance,

in stimulated emission depletion (STED), a technique developed by Hell [9], the effective amount of a material used to register the fluorescence is reduced due to the transmission of some molecules located along the edge of the focused laser radiation light into the nonradiative state (on the wave operation length). Fluorescence depletion is achieved by means of the second laser with a set cross-section beam. In this case, we have to use the technique of confocal scan microscopy, however the basis of the technique is the same. When we register the radiation of a molecule's agglomeration, the size of which is far less than the light wavelength, it is possible to fix the location of this group of molecules with subdiffraction accuracy.

Using STED, PALM, and STORM techniques, we developed the first commercial optical "nanoscopes", i.e. ultrahigh resolution microscopes equipped with laser sources, ultra-sensitive detectors, and powerful computers which help distinguish and restore images quickly. The developed techniques and tools have proved to be in high demand in biochemistry, biophysics, and related sciences. Besides, they open new prospects in the field of nanosensory technology, medical physics, materials science, quantum optics, and nanotechnology.

21.2.2 *Current state of the problem of single point radiators image processing in nanoscopy*

As is pointed out in [15], the processing stage when forming microscopic images, which have ultrahigh resolution and which are obtained with the help of STED, PALM, and STROM techniques, is especially significant because it influences the validity of the obtained results, opportunity to analyze certain images (when the density of marks is high), data analysis delay, compatibility with the used equipment, etc. There is a variety of free and commercial software which have different opportunities. It is noted that if the marks in the unprocessed images are located far from each other, the algorithms to process single marks can be rather accurate and effective. These techniques are implemented in such software as rainSTORM, rapidSTORM, palm3d, and QuickPALM. If input images contain a lot of overlapping peaks, it is necessary to use programs which include certain deconvoluting algorithms, for example, DAOSTORM and

deconSTORM. Besides, the above-mentioned programs require various threshold values to be set. The wrong choice of them can cause significant quality deterioration of the forming image. The situation can be made worse by the fact the characteristics of the forming image can be changed during the experiment.

The comparative analysis of the parameters of the most common programs is performed in the work [11]. It was shown that the best accuracy of the localization and high-speed performance can be provided by software developed on the basis of RapidStorm algorithm (7.6 nm). The implementation of WaveTracer algorithm has a comparable accuracy (7,9 nm) providing the best percentage of the correctly localized objects (93,5%). However, other techniques give less applicable results (87.5% by RapidStorm and 76.5% by QuiqPALM). It should be noted that the obtained results on the localization accuracy depend on the number of marks in the shot. It occurs due to feedback malfunction in the WaveTracer algorithm or ambiguity in the set size choice of the input data both for WaveTracer and RapidStorm. It is obvious that the use in the calculations of a hard-sectored scene, the number of iteration steps, and wavelet transform parameters is bound to the primary signal formation equipment (source — microscope — optical system — video camera).

Due to the fact that the developers of the abovementioned programs do not provide any explanation for the choice of the given parameters, we can conclude that the development of a well-reasoned mathematical approach suitable for their choice is an important task which will let synthesize universal algorithms which allow to obtain a higher percentage of a correct localization at a comparable or better processing time.

21.3 Evaluation of the Statistical Background Characteristics

The information about the statistical background characteristics is necessary to solve the tasks of point marks' detection and when modeling nanoscopic images. We placed emphasis on the fragments which did not have signal marks to evaluate the statistical characteristics in the nanoscopic images. For these fragments, bar graphs were built and the evaluation of the moment of distribution was

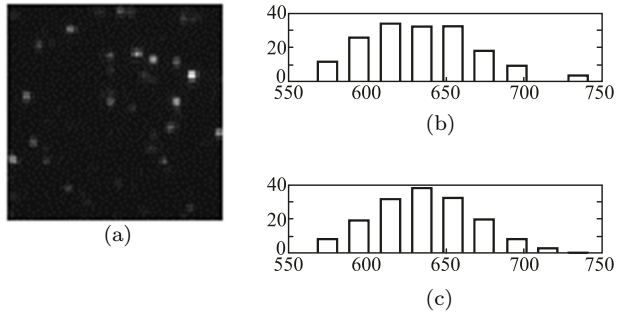


Figure 21.1. An example of nanoscopic image (a), experimental (b), and theoretical (c) bar graphs.

obtained. Figure 21.1 shows an example of a nanoscopic image (fluorescent images of single colloidal semiconductor quantum points [4, 9, 18, 19, 21]).

The hypothesis test about the statistical law showed that if a significance level equals 0.01, the hypothesis about normal distribution is not rejected. Due to this fact, the solution of the following tasks (detection, resolution, and parameters evaluation) can be based on the use of matched filtering, minimization of the root-mean-square errors, etc.

21.4 Evaluation of the Parameters of the Instrumental Response Function of the Nanoscopic Imaging Device

Instrumental response function used for nanoscopic image processing is regarded as a data recording system response which is formed during the single point mark image processing. In its essence, the instrumental response function is an analog of a module of a system impulse characteristic and understanding this function is essential for appropriate solutions of nanoscopic image processing tasks. Theoretically, it was shown that instrumental response function in case of nanoscopic image formation can be described by the Airy function [1], but its application is not convenient. Due to this fact, in some practically important cases, a bidimensional gaussoid can be used instead of it providing sufficient accuracy [1]. The evaluation of

the instrumental response function is theoretically possible on the base of the image of each point mark, however, in practice, the overlapping of different mark responses can be considered as a serious obstacle.

Due to this fact, in this chapter, we offer an approach to define the parameters of the instrumental response function using an autocorrelation function of an image which characteristics uniquely correlated to the instrumental response function in the neighborhood of the principal maximum.

If the instrumental response function does not vary within the shot, the score to the resulting image of each mark with coordinates x_n, y_n can be described in the following way:

$$f_n(x,y) = f_0 + \frac{A(x_n,y_n)}{\Gamma}\exp\left\{-\frac{(x-x_n)^2+(y-y_n)^2}{\Gamma^2}\right\},$$
$$n = 0,1,\ldots,N-1, \tag{21.1}$$

where $A(x_n, y_n)$ is the mark response amplitude, Γ is the width of the instrumental response function of the point source, f_0 is a noise component, and N is the total number of marks in a scene. The resulting image is formed as a sum of single marks responses:

$$f(x,y) = \sum_{n=0}^{N-1} f_n(x,y). \tag{21.2}$$

The autocorrelation function of such an image in the neighborhood of the main peak can be described with the help of the following ratio:

$$\begin{aligned} R(x,y) &= \sum_{n=0}^{N-1} R_n(x,y) \\ &= \sum_{n=0}^{N-1} \frac{A^2(x_n,y_n)}{\Gamma^2}\exp\left\{-\frac{x^2+y^2}{2\Gamma^2}\right\} + f_0^2\cdot X\cdot Y \\ &= f_0^2\cdot X\cdot Y + \exp\left\{-\frac{x^2+y^2}{2\Gamma^2}\right\}\sum_{n=0}^{N-1}\frac{A^2(x_n,y_n)}{\Gamma^2} \\ &= \frac{A^2}{\Gamma^2}\exp\left\{-\frac{x^2+y^2}{2\Gamma^2}\right\} + f_0^2\cdot X\cdot Y. \end{aligned} \tag{21.3}$$

The most important required parameter is the width of the instrumental response function Γ, whereas parameter value A is a scale multiplier which does not influence its pattern.

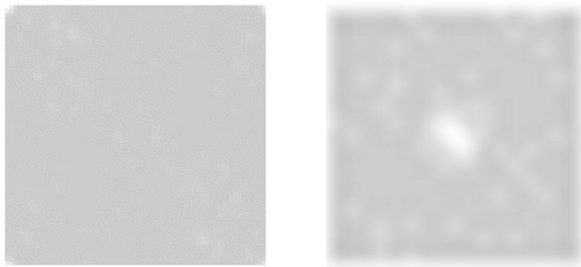
To obtain the numerical values of the instrumental response function width, noise element, and amplitude of the main peak of the auto-correction function we used the Levenberg–Marquardt algorithm [8,11,14]. We obtained the following values of the instrumental response function for the shown in Figure 21.2 image: $\Gamma = 0,786$, $f_0 = 749,46$.

21.5 Detection of Point and Small-Size Objects

The images of the fluorescent marks in a nanoscopic image are the sets of point or small-size objects' parameters which can be defined with the help of the instrumental response function. Due to the fact, as has been shown before, the parameters of the instrumental response function can be estimated in advance, the detection of the point marks can be reduced to the solution of the task of signal detection which would have a known shape against the noise background [5,19].

The procedure itself represents a matched filtering of a signal and detection of the references which exceed the limit set according to the required values of the probability of a false alarm. The limit value is defined as a quantile of a normal distribution.

$$F(U_{c\Phi}) = \frac{1}{\sqrt{2\pi}\sigma_{c\Phi}} \exp\left\{-\frac{U_{c\Phi}^2}{2\sigma_{c\Phi}^2}\right\},$$



(a) (b)

Figure 21.2. General view of the auto-correction function of a nanoscopic image and a detailed imaging of a main peak a nanoscopic image (a) and (b).

where $\sigma_{c\Phi} = \sigma_c \|\mathbf{H}\|$ is a root-mean-square error of noise after matched filtering, $U_{c\Phi}$ is the signal reference mark on the matched filtering output, $\|\mathbf{H}\|$ is a rate of the pulse feature of the matched filter, and σ_c is a root-mean-square error of noise in the original frame.

Figure 21.3 shows the results of matched filtering of the original frame. The check of the distribution law of background reference marks shows that it does not contradict the normal distribution law. Figure 21.3(b) shows the results of the objects' detection in the processed frame at different values of the false alarm probabilities. As can be seen in the given results, the proposed approach provides a successful peak marking in the nanoscopic images. The tasks of their separation and parameters' estimation are solved during the next processing stages.

21.6 Resolution of Point Marks

When solving a task of point marks' resolution, it is necessary to define which marks and what mark parameters caused certain peaks

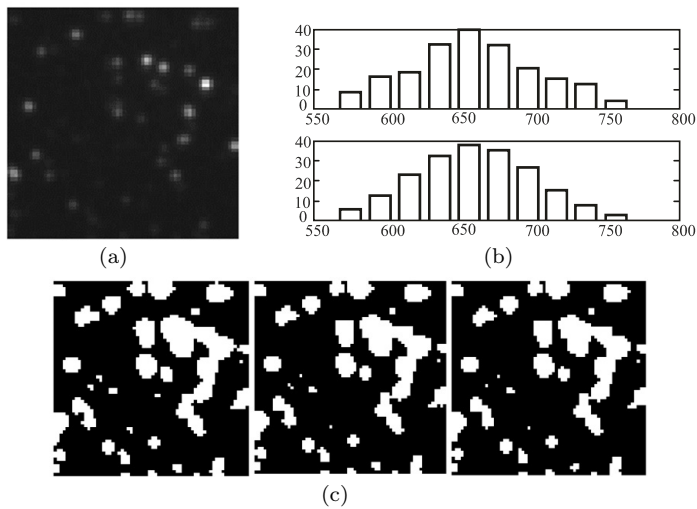


Figure 21.3. Detection of the point and small-size objects: (a) result of the matched filtering of an image; (b) bar graph for background reference marks distribution; (c) the detection result in case of a false alarm probability $p_{\pi m} = 10^{-8}$, $p_{\pi m} = 10^{-9}$, $p_{\pi m} = 10^{-10}$.

in a nanoscopic image, i.e. as a result it is necessary to define the location and the amplitudes of the resolvable marks. Nowadays, various approaches are being used for signal resolution. It is possible to use special kinds of modulation which provide close to delta processing results with the help of matched filtering. However, it is difficult to apply extra kinds of modulation in nanoscopic images. Good results can be obtained with the help of inverse filters, but in this case, reference signal must have a wide quasiuniform or uniform spectrum. The gaussoid which is used for instrumental response function approximation does not meet the requirements. Close to the resolution tasks can be considered a deconvolution task [4, 9, 17, 18, 20–22, 25]. However, there are no universal and clear recommendations on a certain way of deconvolution suitable for all applications. The choice of a deconvolution method usually depends on the developer's experience, requirements for the accuracy and performance, sensitivity to various kinds of interference, and available hard and software. In the works related to nanoscopic image processing, it is recommended to use their approximation with the help of Gaussian bivariate function and parameters adjustment (the number of peaks, location, amplitude) with the help of the Levenberg–Marquardt algorithm for the peak resolution in nanoscopic images. As a resolution quality criterion, it is used as an approximation error term. The variant of peak location with a minimum error is used as a resolution result. The drawback of this method is sensitivity to the edge effects, dependence of the result on the initial conditions, ambiguity of the obtained solution, relatively high computing intensity, and substantial variance of the required for peak resolution time. Due to these factors, we have offered an original approach to the solution of the mark resolution task in nanoscopic images which can be presented in the form of an algorithm given in Figure 21.4.

Approximation of the peak references for subtraction of a gaussoid at each iteration of the algorithm is calculated according to the formula

$$f(x,y) = A(X_{\max}, Y_{\max}) \exp\left\{-\frac{(x - X_{\max})^2 + (y - Y_{\max})^2}{\Gamma^2}\right\} + f_0,$$

where $X_{\max}$, $Y_{\max}$ are the coordinates of the found in the image peak, $A(X_{\max}, X_{\max})$ is a peak amplitude, and Γis a previously found parameter of the instrumental response function. Figure 21.5 shows

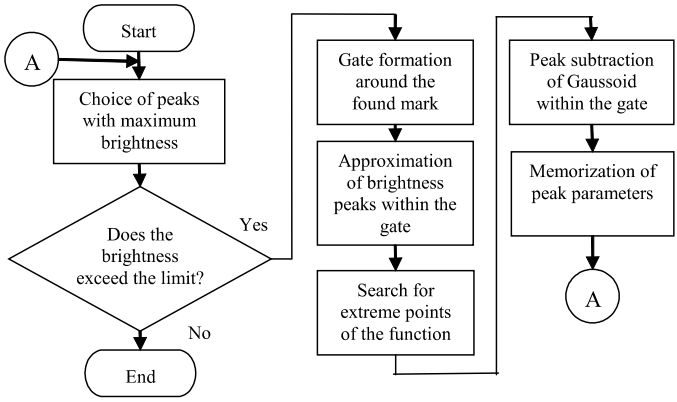


Figure 21.4. Algorithm for point marks resolution in a nanoscopic image.

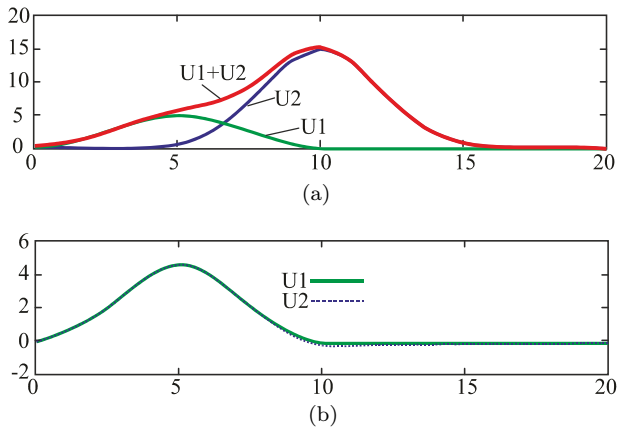


Figure 21.5. Example of interrupted signals resolution: (a) diagram of the mutual location of the signals and their sum; (b) diagram of a signal with the lower amplitude after subtraction of a signal with the higher amplitude.

the work of the peak deconvolution method using the example of a one-dimensional diagram.

The advantages of the given method are its simplicity, linear dependence of labor intensity on the number of marks in the scene, and possibility of parallelizing separate stages of the algorithm. Figure 21.6 shows an example of deconvolution with the use of the method considered above.

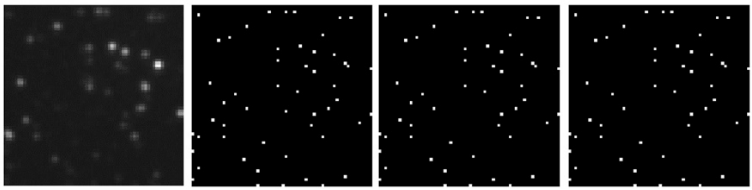


Figure 21.6. Example of the marks deconvolution in a nanoscopic image when the false alarm probability equals $p_{\pi m} = 10^{-8}, p_{\pi m} = 10^{-9}, p_{\pi m} = 10^{-10}$.

21.7 Evaluation of Parameters of Point Marks

The most important marks taken into consideration during the processing of nanoscopic images are peak coordinates x_n, y_n and their amplitudes$A(x_n, y_n)$. Besides, a noise component f_0 and peak width Γ are also evaluated. When using numerical techniques based on the Levenberg–Marquardt algorithm, initial parameter values are set according to the information obtained during the solution of the tasks of statistical background characteristics evaluation, instrumental response function evaluation, detection, and resolution. If the distance from the point mark, for which the parameter evaluation procedure is performed, to the nearest mark exceeds a certain limit, the mark is considered to be isolated and we can neglect the influence of other mark responses when evaluating its parameters. In this case, for approximation, it is necessary to use a single Gaussian function. If the mark is not isolated, the approximation is performed for the sum of the system functions of all marks

$$f_{\text{stroba}}(x, y) = \sum_{n=0}^{N_{\text{stroba}}{}^{-1}} f(x_n, y_n).$$

The calculating procedure of approximating function parameters evaluation is based on the Levenberg–Marquardt algorithm. Let us assume that the point brightness within the gate is set by the references $F(x, y)$. As a result of value parameters' evaluation which set function $f_{\text{stroba}}(x, y)$, the composed function must be minimized [5, 8, 14]:

$$\varepsilon = \sum_x \sum_y (F(x, y) - f_{\text{stroba}}(x, y))^2.$$

At the beginning of the work of the algorithm, a vector is set according to the considered above approaches:

$$\mathbf{w} = \left\{ \begin{array}{l} \{A_0, x_0, y_0, \Gamma_0, f_0\}, \quad \{A_1, x_1, y_1, \Gamma_1, f_1\}, \ldots, \\ \left\{ A_{N_{\text{stroba}}-1}, x_{N_{\text{stroba}}-1}, y_{N_{\text{stroba}}-1}, \Gamma_{N_{\text{stroba}}-1}, \right. \\ \left. \quad f_{N_{\text{stroba}}-1} \right\} \end{array} \right\}^T .$$

At each approximation step, a new vector is calculated $w = w + \Delta w$. Incremental vector Δw is defined according to the ratio:

$$\Delta \mathbf{w} = (\mathbf{J}^T \mathbf{J} + \lambda \mathbf{I})^{-1} \mathbf{J}^T (\mathbf{F} - \mathbf{f}_{\text{stroba}}),$$

where $\mathbf{I}$ is the identity matrix,

$$\mathbf{J} = \begin{bmatrix} \frac{\partial \mathbf{f}_{\text{stroba}}(\mathbf{w},0,0)}{\partial \mathbf{w}_1} & \cdots & \frac{\partial \mathbf{f}_{\text{stroba}}(\mathbf{w},0,0)}{\partial \mathbf{w}_R} \\ \vdots & \ddots & \vdots \\ \frac{\partial \mathbf{f}_{\text{stroba}}(\mathbf{w},X-1,Y-1)}{\partial \mathbf{w}_1} & \cdots & \frac{\partial \mathbf{f}_{\text{stroba}}(\mathbf{w},X-1,Y-1)}{\partial \mathbf{w}_R} \end{bmatrix},$$

R is a vector dimension $\mathbf{w}$, X, Y are gate dimension along the vertical and horizontal axes, and λ is a regulation parameter.

21.8 Particle Tracking in Nanoscopic Images

When forming a nanoscopic image, a lot of factors influence the quality characteristics of the obtained results. Those factors can lead to the corruption of obtained data. One of them is called image drift of nanoparticles. It can be caused by deliberate or occasional shift of support where the radiators are located, motion of the environment, fluctuation of the position of the quantum point radiation center due to its imperfection, etc. In this case, the task of measuring the position of quantum points up to nanometers is particularly important [22]. Its solution can become complicated if the distance between the neighboring points is less than a pixel. Consequently, the drift parameters evaluation of both single-point radiators and their groups is important for the solution of the basic tasks of their processing which include detection, resolution, and parameter evaluation.

In this chapter, we consider an approach which can be applied to the solution of the tracking of the single-point radiators on the base of the Hough transform. The application of the Hough transform in this case is due to the possibility to apply a linear model of particle drift during a certain rather long period of time. For example, as the research has shown, in many cases, a linear model of the global drift can be used for a series of up to 20000 frames at a frequency of 10 frames a second. If we observe significant instability in the support positioning mechanisms, the linear model can be used for a series from a few dozen up to several hundreds of frames.

The classical Hough transform was introduced to solve the tracking and parameter evaluation tasks of the charged particles' paths [10]. A particle track was considered to be represented as a set of straight lines' parameters which should be evaluated. It was offered to use a voting procedure in a parameter space where each point votes for all straight lines which can cross this point. As a result, in the parameter space, peaks are formed. The position of the peaks corresponds to the parameters of the lines along which the majority of the points are located. The classical Hough transform was widespread in technical vision systems used to detect straight fragments in images.

Later, there were introduced some modifications of the Hough transform intended to be used for the detection of images of circumferences, ellipses, and objects with arbitrary shapes [3], as well as for the processing of 3D surfaces [13, 23, 24].

21.8.1 *The Hough transform implementation in 3D space*

For the first time, the classical Hough transform was proposed to solve tasks of charged particles processing the fragments which could be represented in the form of straight lines. To eliminate the ambiguity in the coefficients representation, it was proposed to use a trigonometric representation of a straight-line equation:

$$x\cos(w) + y\sin(w) = R, \tag{21.4}$$

where R is the length of the vertical dropped to a line from the origin of coordinates and w is an angle between the vertical to the line and the OX axis which creates a 2D parameter space.

In a 3D space, the line will be described with four parameters which lead to a 4D parameter space. As a result, we have higher requirements for the storage capacity and performance of the computing system.

In [16], it was proposed to use a line in a polar coordinate system, but it does not help us to reduce the dimensions of the parameter space.

To solve the problem of straight lines' detection in 3D space, it is proposed to use two independent accumulator spaces to describe the lines along each of the coordinates and compare the results afterward. Due to the fact that the point mark shifts over the time between the frames and not beyond a few nanometers, when setting the line, we can eliminate the trigonometric form and use a standard recording of the line equation. Such representation is also considered to be more preferable for calculation, as it does not require the calculation of the trigonometric function.

The location of the center of each point radiator in the frame is set with the coordinates $x(t)$ and $y(t)$. The equations of the corresponding lines can be written in the following way:

$$x(t) = a_x t + b_x, \quad y(t) = a_y t + b_y. \tag{21.5}$$

Based on the information given *a priori* concerning the size of the frame and maximum drift speed of the point radiators, we can set the minimum and maximum values of the coefficients in the line equations (21.5):

$$a_{x_{\min}}, \quad b_{x_{\min}}, \quad a_{x_{\max}}, \quad b_{x_{\max}}, \quad a_{y_{\min}}, \quad b_{y_{\min}}, \quad a_{y_{\max}}, \quad b_{y_{\max}}.$$

To fill in the accumulator space, we should iterate over the values of the coefficients a_x, a_y with step $\delta a = (a_{\max} - a_{\min})/N$, where N is a dimension of the accumulator space with the coordinates a and b, and we calculate the coefficient b value for each point in each frame:

$$b_x = x(t) - a_x t, \quad b_y = y(t) - a_y t.$$

For current values a and b, we calculate their indices I_a, I_b in an accumulator space and increment the corresponding cells:

$$A_x(I_{a_x}, I_{b_x}) = A_x(I_{a_x}, I_{b_x}) + 1, \quad A_y(I_{a_y}, I_{b_y}) = A_y(I_{a_y}, I_{b_y}) + 1,$$

where $I_a = a/\delta a, I_b = b/\delta b, \delta b = (b_{\max} - b_{\min})/N$.

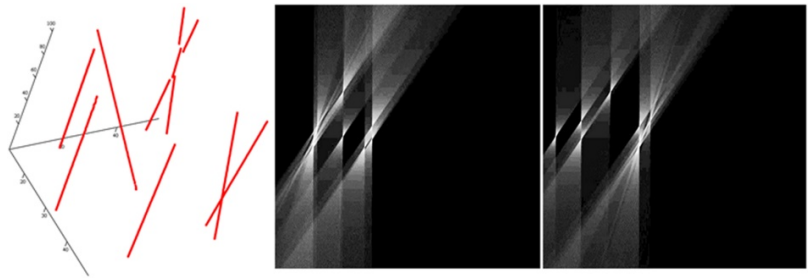
When the search of all the points in all the frames of the accumulator space is finished, we form peak coordinates which help us to evaluate the coefficients a and b values for the tracks which are located in the image (Figure 21.7).

For this purpose, we should perform a linear search of peaks starting with the maximum one. The corresponding coefficient values in the line equation are calculated according to the peak indices in the accumulator space:

$$a = a_{\min} + \delta a \cdot I_a, \quad b = b_{\min} + \delta b \cdot I_b.$$

To reduce the influence of the detected straight line on the processing results of other peaks, we perform a decrement of cells in the accumulator space. Those cells were filled during the processing of the references of this line. As a result, we have a suppression of the found peak and reduction of the background noise level for the other peaks. Afterward, the process of peak detection continues until the references which exceed the set limit are found in the accumulator space.

The next stage of the processing is the comparison of the sets of the coefficients a_x, b_x and a_y, b_y to define whether they belong to the same lines. In this case, the number of coefficients M_x and M_y in the spaces in general can be different as there might be errors related to both not found lines and found false lines. The comparison procedure results in the formation of a 2D space of votes with the dimension $M_x * M_y$ and detection of the number of the nearest points in all frames for each pair of lines. According to the elements of the vote



(a) (b) (c)

Figure 21.7. Example of the Hough transform implementation for tracks in 3D space: (a) image of point radiators tracks; (b) the result of accumulator of space filling for $x(t)$ coordinate; (c) the result of accumulator of space filling for $y(t)$ coordinate.

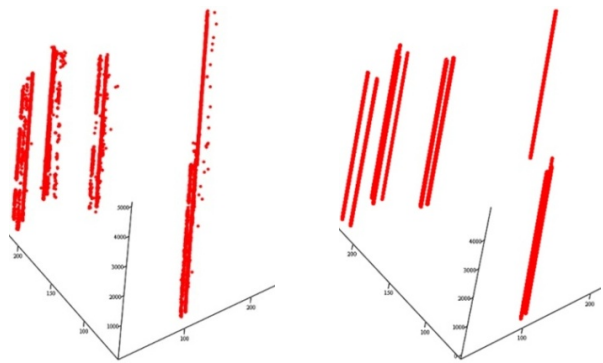
space which exceed the set x limit, we can detect the parameters of the corresponding lines.

21.8.2 *Experimental results*

The experimental research on the efficiency of the proposed technique was performed using the images of the fluorescent microscopy of single colloidal nanocrystals (quantum points) of CdSe/ZnS with the dimension of the radiating nucleus of about 4.4 nm. The terms of the experiment were chosen in such a way that securely isolated from each other images of quantum points could get into the field of regard. To achieve this option, we varied the solution concentration and centrifugal process parameters.

The registration of the fluorescent images in the experimental facility was described in [6]. Figure 21.8 shows the data about the found point marks in separate frames and results of detection compared to their tracks according to the Hough transform.

As the figure shows, the proposed technique provides a good matching of the approximated lines with the experimental data. In this case, the average time spent on the frame processing (when processing a series of 5000 frames, having an accumulator space dimension equal to 500*500, and using an Intel Core-i7 processor and an interpreted programming language without code optimization and without mechanisms for calculation paralleling) is 0.04 second per frame. It provides an opportunity for real-time operation mode.



(a) (b)

Figure 21.8. Example of particle tracks' approximation: (a) location of the found points in frames; (b) tracks' approximation results.

21.9 3D Nanoscopic Images Formation

Improvement of the methods for obtaining images with ultrahigh resolution based on the fluorescent microscopy is a very topical task nowadays. On the one hand, it is due to the ample opportunities to implement it in various applications, for example, for the study of biological objects, membrane materials, semiconductor properties, etc. On the other hand, existing approaches have certain drawbacks, for example, they cannot provide a precise description of 3D objects. In this case, a certain interest is raised in the methods which are based on the formation and processing of double-helix images of single-point radiators (Figure 21.9) [20].

The formation of these images is based on the splitting of the light beam emitted by a single radiator into two beams the mutual position of which definitely describes the third coordinates of the radiator. Nowadays, existing application packages used for the processing of nanoscopic images do not provide complete automation of the parameter evaluation process for double-helix images of the radiators which results in poor accuracy of the radiators' images and long processing time [11, 17, 22].

This chapter deals with the techniques for automatic evaluation of parameters of double-helix images of single-point radiators. The proposed technique in this chapter is based on the interframe correlation of the processed images which is used for the evaluation of parameters and formation of 3D gate path which traces from frame to frame

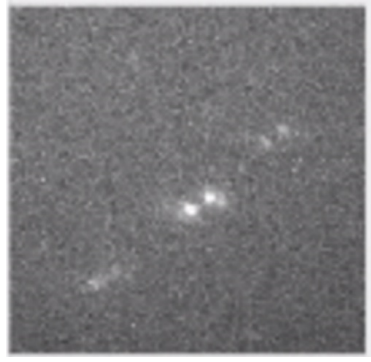


Figure 21.9. Example of a double-helix image.

formed by the image marks 3D helix. The techniques under consideration are a logical development of the technique used to trace the extended objects which are formed by radars with synthesized aperture [7].

21.9.1 *Technique to trace the double-helix images of single point radiators*

The solution of the tasks to trace the path of the double-helix image of radiators must be divided into several stages:

1. *Detection of point radiator images*: The technique to detect double-helix images of point radiators was described in [2] and is based on using the Hoch transform for circles [12]. The result of the Hough transform implementation is the filled accumulator space in the parameter space. The location of the peaks (maximums) in the accumulator space corresponds to the location of the centers, double-helix radiators' images in a frame, while the parameter of the circumference diameter where the peak formation takes place corresponds to the distance between double marks (Figure 21.10). The obtained values are used as input data in the second stage of path tracing.
2. *The gate formation, start of the path tracing*: The gate which is used for path tracing is a virtual cylinder inside which there is a double helix (Figure 21.11). The location of the helix references corresponds to the location in a frame of split image centers of point radiators. The gate boundaries set the area of the frames within which the statistical analysis is performed. This analysis is used for the optimization of the gate parameters. The gate includes several sequentially arranged frames.

The model of a double-helix image of the point marks in n frame can be set with the help of the sum of two Gaussian peaks:

$$f_n(x,y) = f_0 + \frac{A_n}{\Gamma_n}\exp\left\{-\frac{\left(x - x_{y_n} - \frac{D_n}{2}\cos(\varphi_0 + n\Delta\varphi_n)\right)^2 + }{\Gamma_n^2}\right.$$
$$\left.\times\frac{+\left(y - y_{y_n} - \frac{D_n}{2}\sin(\varphi_0 + n\Delta\varphi_n)\right)^2}{1}\right\}$$

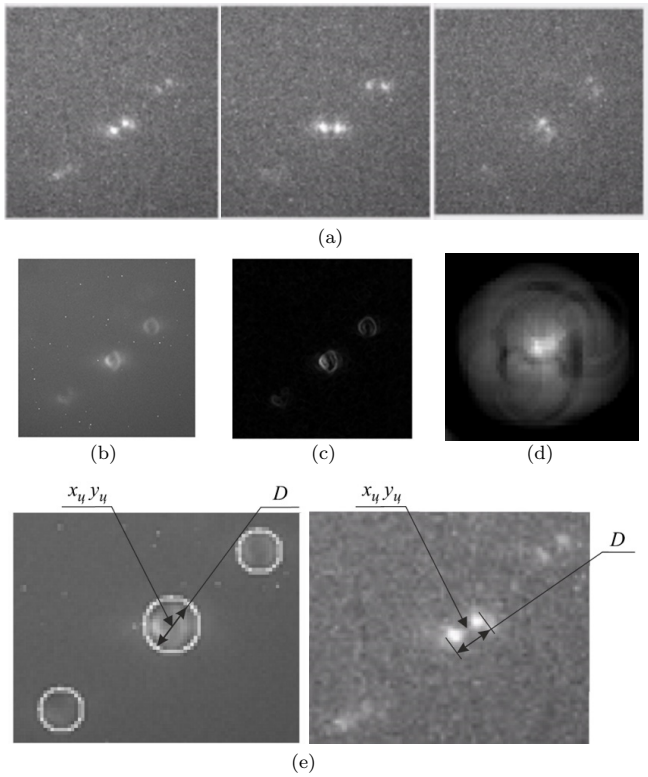


Figure 21.10. Example of the detection and preliminary evaluation of parameters of double-helix image location in a scene based on the Hough transform: (a) a series with "split" images of single points radiators; (b) sum of the series of images; (c) result of the edge underlining; (d) example of the accumulation in the Hough parameter space for right upper mark; (e) the display of the result of circumference detection.

$$+\frac{A_n}{\Gamma_n}\exp\left\{-\frac{\left(x-x_{y_n}+\frac{D_n}{2}\cos(\varphi_0+n\Delta\varphi_n)\right)^2+}{\Gamma_n^2}\right.$$

$$\left.\times\frac{+\left(y-y_{y_n}+\frac{D_n}{2}\sin(\varphi_0+n\Delta\varphi_n)\right)^2}{1}\right\},$$

where A_n is the amplitude of the reference response in n frame, x_{y_n}, y_{y_n} are coordinates of the double-helix image center, Γ_n is the width of the instrumental response function of the point source, D_n is the distance between the mark, φ_0 is the initial rotational angle of the double-helix

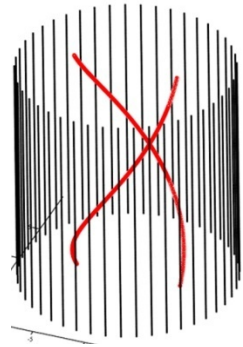


Figure 21.11. The gate view used for tracing the double-helix image of a point radiator.

image in a frame, $\Delta\varphi_n$ is an angle shift in the double-helix image between two neighboring frames, and f_{0_n} is a noise element.

The gate center in the horizontal plane corresponds with the coordinates x_y, y_y which were obtained with the help of the Hough transform, while in the vertical plane, it corresponds with the center of the frames pack. It follows with the initial optimization of gate parameters. To achieve this, it is necessary to use the variability of $x_{y_n}, y_{y_n}, \Gamma_n,\ D_n, \varphi_0, \Delta\varphi_n$ parameters within a narrow range to provide the maximization of the normalizing of scalar product between $f_n(x,\ y)$ function and image references in the analyzed sequence of frames:

$$\eta = \frac{\sum_n \sum_x \sum_y f_n(x,y) \cdot J_n(x,y)}{\|f\|\|J\|} \xrightarrow[\Gamma_n, D_n, \varphi_0, \Delta\varphi_n, x_y, y_y]{} \max . \qquad (21.6)$$

3. *Path tracing*: In the path tracing mode, the gate coordinate increases by 1 at each step in the vertical plane. It is followed by the optimization of function (21.6). The obtained parameter values are memorized and used as a result of path tracing.

Figure 21.12 shows the result of building a path of double-helix images of point marks and evaluation of their parameters.

Proposed in this chapter are techniques for double-helix images' processing of point emitters that provide complex solutions to the nanoscopic image processing tasks: to evaluate parameters of instrumental response function within a certain set of frames, evaluate drift parameters of single radiators, obtain data which are necessary for evaluation of 3D coordinates of point radiators. Moreover, due to the

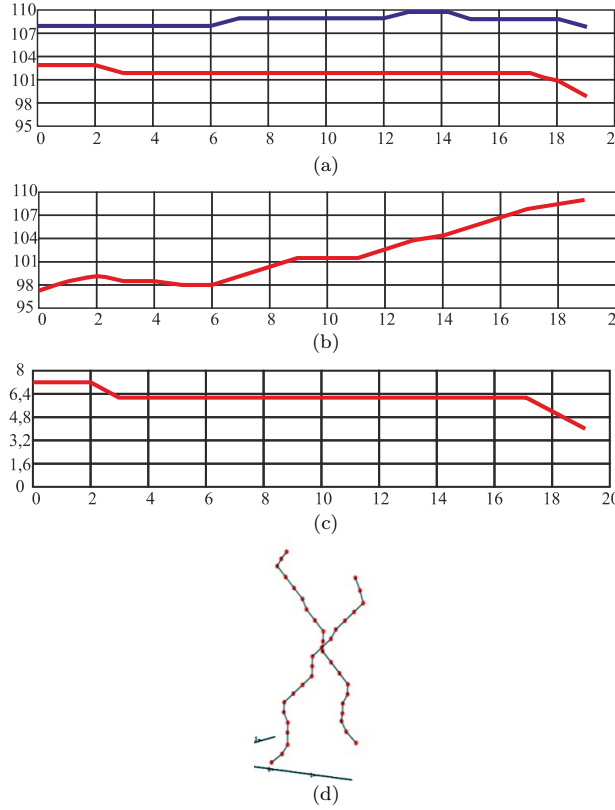


Figure 21.12. The result of parameters' evaluation and path tracking of a double-helix image of a point mark: (a) evaluation of center coordinates; (b) evaluation diagram for angle of rotation; (c) value evaluation diagram; (d) motion of the path of a double-helix image of a point mark.

fact that we take into account data correlation from the neighboring point radiators, we can improve the quality of received evaluations. Automation of data processing allows us to increase the speed of data processing and their accuracy. The analysis of the data processing results based on the double-helix images of semiconductor nanocrystals CdSe/ZnS showed the relevance of the obtained evaluations.

Acknowledgment

We are grateful for the data provided by the research group of the Laboratory of Electronic Spectra of Molecules of the Condensed

Matter Spectroscopy department of the Institute of Spectroscopy of the Russian Academy of Sciences.

References

[1] Abbe, B. (1873) ZurTheorie des Mikroskops und der mikroskopischen Wahrnehmung. *Archivfür Mikroskopische Anatomie*, 9(1), 413–418.

[2] Baev, A.A., Rozhentsov, A.A., Naumov, A.V., and Eremchev, I.Y. (2017) Solution of the task to detect single nanoscopic radiators images in fluorescence microscopy. *XI International Symposium on Photon Echo and Coherent Spectroscopy (FEKS - 2017)*, pp. 126–127.

[3] Ballard, D. (1981) Generalizing the Hough transform to detect arbitrary shapes. *Pattern Recognition*, 13(2), 111–122.

[4] Betzig, E., Patterson, G.H., Sougrat, R., Lindwasser, O.W., Olenych, S., Bonifacino, J. S., Davidson, M.W., Lippincott-Schwartz, J., and Hess, H.F. (2006) Imaging intracellular fluorescent proteins at nanometer resolution. *Science*, 313(5793), 1642–1645.

[5] Demidenko, E.Z. (1989) *Optimization and Regression* (Nauka, Moscow).

[6] Eremchev, M.Y., Eremchev, I.Y., and Naumov, A.V. (2017). Submitted in ACSNano.

[7] Furman, Y.A., Peredreev, A.K., Krevetsky, A.B., Rozhentsov, A.A., *et al.* (2002) *Introduction to Contour Analysis and Its Application to the Processing of Images and Signals.* Furman, Y.A. (ed.) (Fizmatlit, Moscow).

[8] Gill, F., Murray, W., and Wright, M. (1985) *Practical Optimization* (Mir, Moscow).

[9] Hell, W. (2007) Far-field optical nanoscopy. *Science*, 316(5828), 1153–1158.

[10] Hough, P.V.C. (1962) Method and means for recognizing complex patterns, U.S. Patent 3 069 654.

[11] Kechkar, A., Nair, D., Heilemann, M., Choquet, D., and Sibarita, J.-B. (2013) Real-time analysis and visualization for single-molecule based super-resolution microscopy. *PLoS One*, 8(4), e62918. DOI: 10.1371/journal.pone.0062918.

[12] Kimme, C., Ballard, D., and Sklansky, J. (1975) Finding circles by an array accumulators. *Communications of ACM*, 2(Issue 2), 120–122.

[13] Khoshelham, K. (2007) Extending general Hough transform to detect 3D object in laser range data. *ISPRS Workshop on Laser Scanning and SilviLaser*, pp. 206–211.

[14] Levenberg, K.A. (1944) Method for the solution of certain problems in last squares. *Quarterly of Applied Mathematics*, 2, 164–168.

[15] Liangwongsan, S., Marungsri, B., Oonsivilai, R., and Oonsivilai, A. (2011) *Extracted Circle Hough Transform and Circle Defect Detection Algorithm* (World Academy of Science, Engineering and Technology), Vol. 6, pp. 432–438.

[16] Jeltsch, M., Dalitz, C., and Pohle-Frohlich, R. (2016) Hough parameter space regularisation for line detection in 3D. *Proceedings of the International Conference on Computer Vision Theory and Applications (VISAPP)*, pp. 345–352.

[17] Metcalf, D.J., Edwards, R., Kumarswami, N., and Knight, A.E. (2013) Test samples for optimizing STORM super-resolution microscopy. *Journal of Visualized Experiments*, 79, e50579. DOI: 10.3791/50579.

[18] Moerner, E. and Kador, L. (1989) Optical-detection and spectroscopy of single molecules in a solid. *Physical Review Letters*, 62(21), 2535–2538.

[19] Naumov, A.V. (2013) Spectroscopy of organic molecules in solid matrix at low temperatures: From Shpolsky to laser fluorescence spectromicroscopy of all effectively radiating single molecules. *Success in Physical Science*, 183(6), 633–652.

[20] Naumov, V., Eremchev, I.Y., and Gorshelev, A.A. (2014) Laser selective spectromicroscopy of myriad single molecules: Tool for far-field multicolor materials nanodiagnostics. *European Physical Journal D, Colloquium Paper* (in print). DOI: 10.1140/epjd/e2014-50414-x.

[21] Orrit, J.B. (1990) Single pentacene molecules detected by fuorescence excitation in para-terphenyl crystal. *Physical Review Letters*, 65(21), 2716–2719.

[22] Pavani, R.P., Thompson, M.A., Biteen, J.S., Lord, S.J., Liu, N., Twieg, R.J., Piestun, R., and Moerner, W.E. (2009) Three-dimensional, single-molecule fluorescence imaging beyond the diffraction limit by using a double-helix point spread function. *Proceedings of the National Academy of Sciences of the USA*, 106(9), 2995–2999.

[23] Rozhentsov, A.A., Morozovsky, K.V., and Baev, A.A. (2013) 3D generalized Hough transform, invariant to parameters of rotation and scaling. *Telekommunikatsii*, 9, 11–16.

[24] Rozhentsov, A.A., Morozovsky, K.V., and Baev, A.A. (2013) Modified generalized Hough transform to process 3D images with unknown parameters of rotation and scaling. *Avtometria*, 49(2), 30–41.

[25] Rust J., Bates, M., and Zhuang, X.W. (2006) Sub-diffraction-limit imaging by stochastic optical reconstruction microscopy (STORM). *Nature Methods*, 3(10), 793–795.

© 2025 World Scientific Publishing Company
https://doi.org/10.1142/9789811267215_0022

Chapter 22

Models, Methods, and Information Technologies for Creating, Processing, and Analyzing Spatially Distributed Data

Yury G. Vasin, Leonid I. Lebedev*, and Tamara Sh. Utesheva†

The Institute of Information Technology, Mathematics and Mechanics of the National Research Lobachevsky State University of Nizhny Novgorod, Nizhny Novgorod, the Russian Federation

**lebedev@pmk.unn.ru*

†*uts13@yandex.ru*

This chapter presents a review of the results of fundamental and applied research performed by the team of authors.

The following methods and algorithms have been developed for constructive synthesis of the following:

- various orthonormal well-adapted basis functions (WABFs) that satisfy the specified properties and cover the initial set of points with a predetermined accuracy in a given metric. The process of coding reduces to calculating the expansion coefficients for these functions,
- various oblique-angle bases (smoothing (WASBFs) and restoring (WARBFs) functions). The coding of signals reduces to selecting irregularly located and smoothed with the respective WASBFs significant samples of the original signal. Decoding is carried out with the use of WARBFs,
- local homogeneous time-spaced smoothing and restoring functions (short bursts). The procedure for their synthesis is reduced to solving

the multiextremal problem. On the basis of these functions, effective algorithms for adaptive compression and filtering with maximum coordinate error control have been developed. At the output of such algorithms, the signals and the information field are represented as a hierarchical set of significant samples and errors (truncated difference binary trees).

A general-to-specific method has been developed that is based on the proposed hierarchical structures for representation of 2D and 3D information and on decision-making methods for these structures. Intermediate solutions obtained in the course of processing the hierarchy upper levels can first be used as some approximation to the desired solution, and then they are used to evaluate the promising directions for continuing the search for the ultimate solution.

Correlation-extreme contour methods (CECM) for recognition of graphic objects have been developed. These methods are based on the calculation of similarity (proximity) estimates of the object and template contours and are invariant with respect to orthogonal transformations and scaling. The criteria for estimating the similarity are the root-mean-square deviation and the Hausdorff distance. CECM recognition algorithms have been implemented in two modes: learning and self-learning.

An algorithm has been developed for near-lossless compression of large-format raster graphic documents with a weakly formalized description of discrete objects and inscriptions. It is based on a two-criteria CECM algorithm with self-learning.

New intelligent information technologies and software have been developed for vectorizing such documents based on recognition methods with self-learning and computational geometry methods. All the developed algorithms' end technologies are distinguished by novelty, low time consumption, and lowcomplexity, while coding and adaptive compression algorithms offer a high compression ratio.

Keywords: "Adapted" basis, local function, computational geometry, hierarchical structures, hierarchical representation, D-neighborhoods, recognition data, correlation-extreme contour methods, estimate invariant, parametric optimization, experimental testing, the symmetry axes of figures, obtaining consistent descriptions, the global minimum location, parabolic method, structural optimization, intellectual technology, vectorization, symbol recognition on maps, images compression, Hausdorff distance, weakly formalized description of objects.

22.1 Introduction

Here, we present a review of the results of fundamental and applied research performed by the authors' team.

This chapter focuses on methods, algorithms, and technologies that are still in demand in big data analysis and in the processing of spatially distributed information. Updated methods for constructing "adapted" bases have found a new use in the analysis of hyperspectral images.

Local homogeneous smoothing and restoring functions, which are substantially analogous to the Daubechies, Haar, Meyer, and other wavelets, but are better adapted to the type of information, are used to compress large-format raster graphic images.

We also propose the general-to-specific method based on the hierarchical structures for the representation of 2D and 3D information and on some decision-making methods for such structures. Geometrical problems are solved iteratively: the search for a solution begins at the root of the hierarchy. Intermediate solutions, first, can be used as an approximation to the desired solution, and second, they are used to assess promising areas where the final solution can be found.

Recognition of graphic and textual information is based on correlation-extreme contour methods (CECM). The theoretical framework of these methods is described as a set of theorems. These methods belong to the group of methods involving comparison with templates and are based on estimates that are invariant with respect to orthogonal transformations (ε_m^t), orthogonal transformations and scaling ($\varepsilon_m^{\text{ts}}$), and affine transformations ($\varepsilon_m^{\text{at}}$). The complexity of obtaining the first two estimates with consistent descriptions of the template and the object is proportional to the total number of points on the contours. To harmonize the descriptions, a fast method for localizing the global minimum of the function $\varepsilon_m(s)$ is used, where s is the parameter that specifies one of the cyclic contour descriptions. To refine the consistency of descriptions, the parabolic method is used. Unlike the well-known FineReader and CuneiForm methods, these methods can be used to recognize chaotically located and oriented inscriptions in the images that are tens of times larger than a text page and are easily learnable [17, 18]. CECM-based recognition is implemented in three modes: with learning, with self-learning, and with additional learning (adding unrecognized objects to the list of templates). To improve recognition quality, a two-criteria recognition algorithm is used. The second criterion for evaluating the similarity is the Hausdorff distance. To provide a concise description

of large-format raster images, which were obtained using CECM in the additional learning mode, an intelligent IFc format was created. The IFc format is similar to the DjVu format, but the former one uses a more efficient recognition method and offers a wider range of application possibilities.

22.2 Compression of Signals and Images

Let a set of points $\mathbf{x}^j \in \mathbf{X}$ of the space $\mathbf{R}_n$ be given. Let us consider the methods for compressing signals and images when the original description of objects includes hundreds, thousands, and even tens of thousands of features and the sample size can exceed millions of items. The number of description features can be reduced if a basis is obtained in the original space where the whole set $\mathbf{x}^j \in \mathbf{X}$ could be represented with an error not exceeding some threshold value in a given metric. A basis that is formed on the set $\mathbf{x}^j \in \mathbf{X}$ using only its properties, without first specifying its being part of any model, will be called "adapted" to the given type of information.

22.2.1 *Methods for obtaining an "adapted" basis*

The optimal basis that at the same time minimizes the error and the number of vectors of the subspace is the Karhunen–Loeve basis. This basis is formed from the eigenvectors of the covariance (correlation) matrix that correspond to the largest eigenvalues. However, it is extremely difficult (in terms of computation) to obtain it with such dimensions and sample size, because of the possible loss of significant digits or an overflow during arithmetic operations. Therefore, to obtain an "adapted" basis, it is proposed to use a two-stage procedure. At the first stage, we find the basis $\mathbf{W} = (\mathbf{w^1}, \mathbf{w^2}, \ldots, \mathbf{w}^k)$ using the scheme [5, 7]

$$\mathbf{b}^r = \mathbf{x}^r - \sum_{i=1}^{j-1} \left\langle \mathbf{x}^r, \mathbf{w}^i \right\rangle \cdot \mathbf{w}^i, \quad q = \arg\max_{\mathbf{x}^r \in \mathbf{X}} \|\mathbf{b}^r\|, \ \mathbf{z}^j = \mathbf{b}^q,$$

$$\mathbf{w}^j = \frac{\mathbf{z}^j}{\|\mathbf{z}^j\|}, \tag{22.1}$$

and convolution coefficients $y_i = \langle \mathbf{x}, \mathbf{w}^i \rangle$, which are the features for representing the points of the set in the vector subspace of the basis $\mathbf{W}$.

At the second stage, the covariance matrix $\mathbf{S} = \sum_{\mathbf{y}\in\mathbf{Y}}(\mathbf{y}\cdot\mathbf{y}^T)$ is calculated and its eigenvalues λ_i and vectors $\mathbf{e_i} \in \mathbf{E}$, $i = 1, 2, \ldots, k$ are found. Now, the final version for obtaining the "adapted" basis will be given by the formula

$$\mathbf{u^j} = \sum_{i=1}^{k} e_i^j \cdot \mathbf{w^i}. \tag{22.2}$$

We can use the ratio of eigenvalues λ_1 and λ_m, $m \leq k$ as a criterion for choosing the number m of the vectors in the basis $\mathbf{U} = (\mathbf{u^1}, \mathbf{u^2}, \ldots, \mathbf{u^m})$, while the ratio of the norm of the vectors $\mathbf{z_1}$ and $\mathbf{z_k}$ can serve for the choice of the number k of the basis $\mathbf{W}$. If the resulting basis $\mathbf{U}$ does not satisfy the quality requirements for the description of the set $\mathbf{X}$ points, then the above procedure for generating the vectors of the bases $\mathbf{W}$, $\mathbf{E}$, and $\mathbf{U}$ is repeated for recovery errors in order to find an additional basis. It should be noted that the Euclidean, Chebyshev, and Mahalanobis distances and their combinations can be used as a metric when finding the "adapted" basis. The coding process from the found orthonormal vectors of the "adapted" basis reduces to the calculation of the expansion coefficients in these functions.

22.2.2 *Methods for obtaining a set of informative features. Bases* K *and* G

The image $\tilde{\mathbf{x}}$ of any point $\mathbf{x} \in \mathbf{X}$ can be represented either by expansion coefficients $c_1, c_2, \ldots, c_m$ in the subspace of the basis $\mathbf{U}$, or by coordinates values in the space $\mathbf{R}_n$:

$$\tilde{\mathbf{x}} = \sum_{j=1}^{m} c_j \cdot \mathbf{u}^j = \sum_{j=1}^{m} \left\langle \mathbf{x}, \mathbf{u}^j \right\rangle \cdot \mathbf{u}^j = \sum_{i=1}^{n} x_i \cdot \boldsymbol{\gamma}^i. \tag{22.3}$$

Now, we introduce the matrix $\boldsymbol{\Gamma} = (\boldsymbol{\gamma}^1, \boldsymbol{\gamma}^2, \ldots, \boldsymbol{\gamma}^n)$. According to the definition of the vectors $\boldsymbol{\gamma}^i$, the matrix $\Gamma = \mathbf{U}\cdot\mathbf{U}^T$. It follows from this representation that the rows of the matrix $\mathbf{U}$ are the expansion coefficients of the vectors $\boldsymbol{\gamma}^i$ over the basis $\mathbf{U}$. Since the rank of the matrix $\mathbf{U}$ equals m, then, denoting linearly independent $\boldsymbol{\gamma}^i$ by means of $\mathbf{g}^j$, we will have $\mathbf{G} = (\mathbf{g}^1, \mathbf{g}^2, \ldots, \mathbf{g}^m)$. Linearly independent rows of the matrix $\mathbf{U}$ with the numbers $\nu_1, \nu_2, \ldots, \nu_m$ correspond to the vectors $\mathbf{g}^j$. We use them to form a matrix $\mathbf{V}$. Then, $\mathbf{G} = \mathbf{U}\cdot\mathbf{V}^T$.

Hence, if we use $\mathbf{K} = (\mathbf{k}^1, \mathbf{k}^2, \ldots, \mathbf{k}^m)$ to denote the matrix of the expansion coefficients of vectors $\boldsymbol{\gamma}^i$ over the basis $\mathbf{G}$, then $\boldsymbol{\Gamma} = \mathbf{K} \cdot \mathbf{G}^T$ and, consequently, $\mathbf{K} = \mathbf{U} \cdot \mathbf{V}^{-1}$ [1]. Based on the obtained results, we can write the following formulas for restoring and smoothing the components of the original points

$$\tilde{\mathbf{x}} = \sum_{j=1}^{m} \tilde{x}_{v_j} \cdot \mathbf{k}^j, \cdot \tilde{x}_i = \left\langle \mathbf{x}, \boldsymbol{\gamma}^i \right\rangle, \quad \tilde{x}_{v_j} = \langle \mathbf{x}, \boldsymbol{\gamma}^{v_j} \rangle = \left\langle \mathbf{x}, \mathbf{g}^j \right\rangle. \quad (22.4)$$

In the absence of noise, the coding of signals (point descriptions) reduces to the selection of irregularly located significant samples of the original signal (informative features) whose numbers are determined by the numbers of linearly independent vectors of the smoothing basis $\mathbf{G}$, while in the presence of noise, these are determined by their smoothed values. Decoding is carried out using the restoring basis $\mathbf{K}$.

22.2.3 *Local homogeneous smoothing and restoring functions*

To compress signals and other experimental curves described by a large number of samples N, special methods, and algorithms were created based on the use of local smoothing and restoring functions. To obtain these functions, a set of fragments with a constant description dimension $n \ll N$ was generated on the basis of the initial curves. From this set of fragment, an "adapted" basis $\mathbf{U}^1$ was found, and from it, the bases $\mathbf{G}^1$ and $\mathbf{K}^1$ were found for filtering and restoring the samples as well as the number of informative samples. The dimension n of fragment description and the number $m = 2k$ of bases were chosen based on the following condition: $n = 2m + 1$. Curve compression at the first level now consisted of replacing the description of the current fragment with a set of informative samples (smoothed or initial), if the sample recovery error over the basis $\mathbf{K}^1$ did not exceed the specified threshold. Otherwise, the description of the fragment and its status did not change. At the second level of compression, only first-level fragments were considered. Compression was carried out using the bases $\mathbf{U}^2$, $\mathbf{G}^2$, and $\mathbf{K}^2$ obtained from the first-level fragments. In the same way, compression was performed at subsequent levels [6].

For local homogeneous functions **G** and **K**, we first assign the numbers to informative samples by declaring that such numbers are all the even features of the fragment. Next, we obtain the central function γ^c, $c = (n+1)/2$, which is not linearly independent. However, by using this function, one can get a smoothed value of the central sample for any fragment. Let us replace all basis functions **G** with a vector γ^c by matching its central point with the numbers of linearly independent samples and complementing their descriptions with zeros. As a result, we obtain m vectors from which we obtain the basis **K** that enables restoring the fragment's central sample. Thus, only one function γ^c is used for smoothing, while the same set of coefficients $(k_c^1, k_c^2, \ldots, k_c^m)$ obtained for this function is used in the restoring process. Based on the results obtained, various compression algorithms have been implemented, including recurrent adaptive compression algorithms and adaptive compression algorithms that employ initial information thinning [9].

22.3 Computational Geometry Based on Hierarchical Structures

In [6–12, 14], methods and algorithms for processing graphic information based on local homogeneous well-adapted basis functions (LHWABF) and hierarchical regular structures for data representation in the form of truncated difference binary trees have been proposed.

The samples of each hierarchy level of the above-mentioned structures are selected using the LHWABF and represent the initial data with some controlled error. The lower the level of data presentation hierarchy, the more accurate the initial field description. As a whole, the hierarchical structure contains a non-redundant set of significant samples and description errors at previous hierarchy levels, allowing for the restoration of the original information with a given accuracy. This structure can be represented in the form of truncated binary difference trees (TBDT) for various types of graphic data.

The hierarchical representation of graphical information makes it possible to effectively use the general-to-specific method to analyze the relative position of cartographic objects. Geometric problems are solved by successive intermediate approximations: the search for a

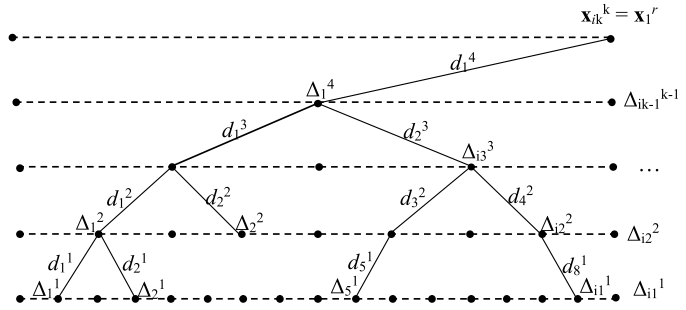


Figure 22.1. Truncated difference tree with D-neighborhoods.

solution begins at the root of the hierarchy and ends with a possibly more coarser data representation. In this case, intermediate solutions are formed at the upper levels of the tree, which, first, can be used as some approximation to the desired solution, and, second, they are used to assess promising areas for further search for the final solution.

When implementing such an approach, it is important to make a correct choice of selection criteria for promising sections of approximation to make sure that the desired solution is obtained by refining them. To obtain such criteria when solving a wide class of problems, restrictive estimates are required at each vertex of the tree at each level. For solving problems of graphic data analysis, it was proposed to use truncated binary difference trees with D-neighborhoods.

D is a neighborhood of the level k, D_i^k is a region beyond which the initial information cannot deviate from the ith approximating section of the kth level.

Figure 22.1 shows a truncated difference tree with D-neighborhoods of the hierarchical representation of a curve on a plane. Here, $\mathbf{x}_i$ are the samples of the vertices of the upper level of the curve approximation, Δ_i^k are coordinate-wise recovery errors. In this case, the graphical information is represented in the form of truncated binary difference trees (truncation is both from above and below). The roots of the trees contain significant samples of the kth rank, and the subordinate peaks contain the values of errors (differences) of the significant samples of the $(k-1)$th, $(k-2)$th, and 1st ranks.

When using for approximation the coefficients of local well-adapted restoring basis functions $k_m(\boldsymbol{\alpha}_1, \boldsymbol{\alpha}_2, \ldots \boldsymbol{\alpha}_m)$, $\sum_{i=1}^m \boldsymbol{\alpha}_i = 1$,

with $m = 2k(\alpha_1, \alpha_2)$, where $\alpha_1 = \alpha_2 = 0.5$, the following law of coordinate restoration (piecewise linear approximation) holds for the samples at the vertices of each level of the tree:

$$v = i^k, \quad \lambda = 2^{k-1}, \quad \mathbf{x}_v^k = 0.5\mathbf{x}_{i-\lambda}^k + 0.5\mathbf{x}_{i+\lambda}^k + \Delta_v^k, \quad 1 \le k \le r$$

where

$$i = \begin{cases} 2^{k-1} \cdot i^k + 1, \text{ for } i^k = 1, k \ge 2, \\ 2^{k-1} \cdot \left(2i^k - 1\right) + 1, \quad \text{for} \quad i^k \ge 2, k \ge 2. \end{cases}$$

Here, $\mathbf{x}_v^k$ is the sample vector with the coordinates (x_v^k, y_v^k).

D-neighborhoods for the case being considered are represented as follows. Each vertex of the tree $\mathbf{x}_v^k$ is associated with two values $d_v^{k,1}$ and $d_v^{k,2}$ that are equal to the maximum distance from the points of the original curve located between the samples $\left[\mathbf{x}_{i-\lambda}^k, \mathbf{x}_v^k\right]$ and $\left[\mathbf{x}_v^k, \mathbf{x}_{i+\lambda}^k\right]$ to the segments specified by these samples.

$$d_v^{k,1} = \max_j \left\{R\left(\mathbf{x}_j, \left[\mathbf{x}_{i-\lambda}^k, \mathbf{x}_v^k\right]\right)\right\}, \quad d_v^{k,2} = \max_j \left\{R\left(\mathbf{x}_j, \left[\mathbf{x}_v^k, \mathbf{x}_{i+\lambda}^k\right]\right)\right\},$$

$$j = (i - \lambda + 1), (i - \lambda + 2), \ldots, (i - 1),$$

where R is the distance between the point of the initial curve and the approximating segment.

The corresponding regions are the D-neighborhoods of the segments introduced earlier (Figure 22.2). Thus, the initial data are represented in the form of a hierarchy of approximating segments and the corresponding D-neighborhoods, which allow us to obtain an increasingly accurate description of the information by going down the structure.

This makes it possible to organize calculations using the general-to-specific principle.

With the use of the general-to-specific approach, based on the hierarchical data representation structures, algorithms were developed for determining the following: relative positions of points and curves; relative positions of curves; relative positions of curves and regions; relative positions of regions; sections of the metric of linear and area objects when the distance between such objects does not exceed some specified threshold.

The algorithms developed have a number of important advantages: high computational performance; a common methodological

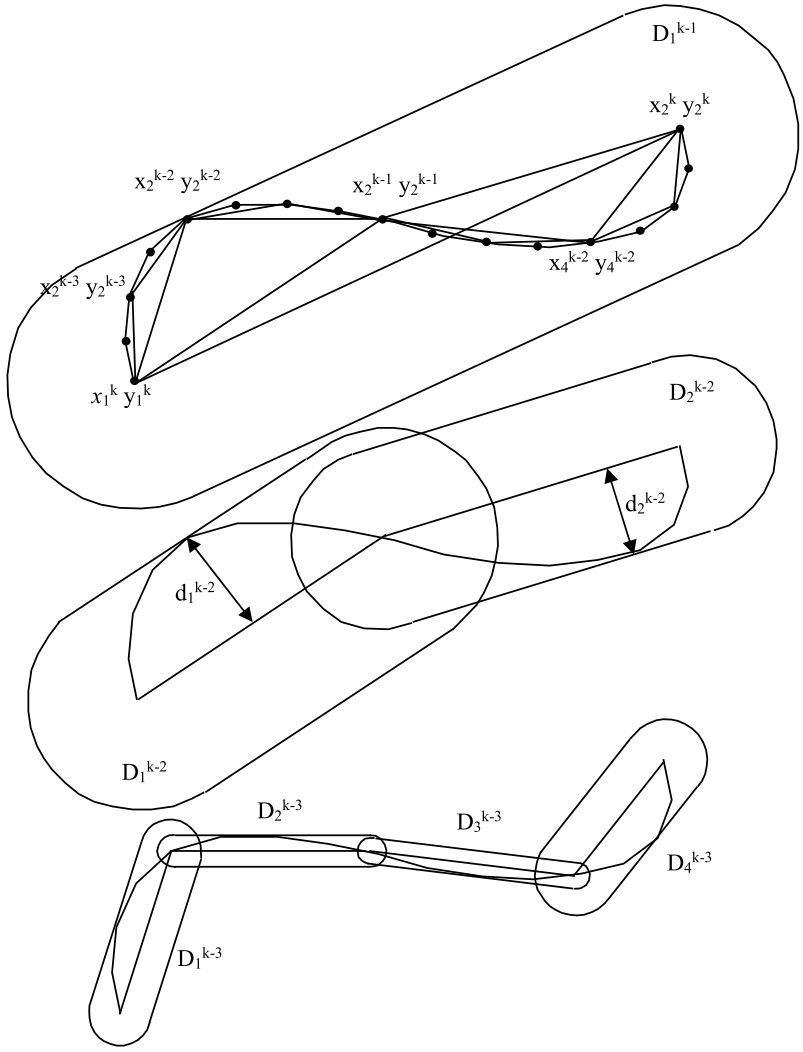


Figure 22.2. The hierarchy of approximating segments and the corresponding D-neighborhoods.

approach to solving all geometric problems; the use of unified data preprocessing, which is carried out once and is used to solve all the necessary tasks while being part of the general technology for graphic information processing.

For each task, the selection criteria for S promising sections of the hierarchical information representation are defined. By refining them, the solution of the problem is ensured. The selection criterion of promising sections with the corresponding values of A_i, B_i, whose refinement will lead with certainty to the desired solution, has the form $S : A_j \leq \min\{B_j\}$ and is true for the following algorithms: calculation of the distance from a point to a curve; search for the curve from a set closest to the given point; search for the point closest to the given curve from a set of points; calculation of the distance between curves on a plane; calculation of the distance from a point to a surface; calculation of the distance between a surface and a curve; calculating the distance between two surfaces.

Here, $A_i = \max\{0, r_i - d_i\}, B_i = r_i + d_i$ or $B_i = r_i$, d_i is the dimension of the D-neighborhood of a segment or a flat element in R^3 of rank k, and r_i is the distance between approximation elements of two objects.

The algorithm for determining the intersection of a ray with a curve has a criterion for selecting approximation sections that are subject to further reconstruction. This criterion has the following form:

$$S : [(p, q), \infty] \cap [(x_i^k, y_i^k), (x_{i+1}^k, y_{i+1}^k)] \vee (h < d); \quad 1 \leq k \leq r,$$

where h is the distance from the approximating segment $[(x_i^k, y_i^k), (x_{i+1}^k, y_{i+1}^k)]$ to the ray, d is the dimension of the D-neighborhood of the rank k segment, r is the maximum depth of the truncated binary difference tree. For the algorithm for determining the position of a point relative to the region:

$$S : (y_{\min} - d < q < y_{\min}) \vee (y_{\max} < q < y_{\max} + d) \vee (h < d),$$

where h is the distance from the point under consideration to the approximating segment of the region's boundary, $x_{\min}$, $x_{\max}$, $y_{\min}$, $y_{\max}$ are extreme coordinates of the approximating segment, d is the characteristic value of its D-neighborhood.

For the algorithm that determines the junction areas of two curves:

$$S : R \leq eps + d_1 + d_2,$$

where R is the distance between the approximating segments of the leading and the guided curve, d_1 and d_2 are the dimensions of the corresponding D-neighborhoods of these segments, *eps* is the junction

threshold. For the algorithm for determining the coincidence sections and the intersection points of two curves, the criterion for selecting approximation sections to be reconstructed has the form:

$$S : \{(h_1 \cdot h_2 < 0 \wedge h_3 \cdot h_4 < 0) \vee r_i < d_1 + d_2\},$$

where h_1, h_2, h_3, h_4 are the distances from the end points of the approximating segment of a curve to a straight line passing through the approximating segment of another curve, $r_1(i = 1, 2, 3, 4)$ are the distances from the endpoints of an approximating segment to another one, d_1 and d_2 are the dimensions of the corresponding D-neighborhoods of these segments. For the algorithm for determining the parts of the structured curves that lie within the region:

$$S : \{(h_1 \cdot h_2 < 0 \wedge h_3 \cdot h_4 < 0) \vee r_i < d\},$$

where h_1, h_2, h_3, h_4 are the distances from the end points of the approximating segment of the curve to the straight line passing through the segment of the region's boundary, $r_1(i = 1, 2, 3, 4)$ are the distances from the end points of the approximating segment of the curve to the segment of the region's boundary, d is the dimension of the corresponding D-neighborhood of the approximating segment of the curve. The algorithm for determining the parts of a structured curve that fall inside a region whose boundary is defined by a structured curve has a selection criterion similar to that of the algorithm for determining the coinciding sections and the intersection points of two curves.

A software package for solving computational geometry problems related to the analysis of the relative positions of objects on a plane has been implemented. The programs of the package have the form of functions and are organized as a library. The superposition of basic functions allows us to solve a very wide class of problems to determine the geometric and logical relations between graphic objects. The software suite is implemented in the programming language C++ Builder.

Under real operational conditions of the software system in various organizations, the effectiveness of the approaches adopted for developing the methods and algorithms.

22.4 Recognition of Graphic and Text Data

This section outlines the basics of correlation-extreme contour methods (CECM), which are key for recognizing graphic and test images. These methods belong to the group of methods involving comparison with templates and are based on the contour model of image representation. As a criterion of similarity (closeness) of forms, the root-mean-square estimate obtained by matching the contours is used. The choice of this estimate is determined by the least squares method.

22.4.1 *Basics of correlation-extreme contour methods*

Let the contour of the template $\mathbf{E}$, of length S^e, be represented on the plane XOY by a sequence of points $\mathbf{w}^{\boldsymbol{e}} = \{\mathbf{w}_{\mathbf{1}}^{\boldsymbol{e}}, \mathbf{w}_{\mathbf{2}}^{\boldsymbol{e}}, \ldots, \mathbf{w}_{\mathbf{m}}^{\boldsymbol{e}}\}$, specified in a Cartesian coordinate system, where $\mathbf{w}_{\boldsymbol{i}}^{\boldsymbol{e}} = (x_i^e, y_i^e)^T$. Let us define the following model for producing objects of the equivalence class $K(\mathbf{E}, \mathbf{G})$, generated by the template $\mathbf{E}$ and the affine transformation $\mathbf{G}$. First, we transform the description of the template contour $\mathbf{w}^{\boldsymbol{e}}$ in accordance with a given affine transformation group $\mathbf{G}$

$$\hat{\mathbf{w}}^{\boldsymbol{e}} = \mathbf{A} \cdot \mathbf{w}^{\boldsymbol{e}} + \Delta \mathbf{w}^{\boldsymbol{e}}, \tag{22.5}$$

where $\boldsymbol{\Delta}\mathbf{w}^{\boldsymbol{e}}$ and $\mathbf{A}$ are the displacement vector and the affine transformation matrix, respectively. Next, we carry out the procedure of imposing noise with a normal distribution $N(0, \sigma)$ on the coordinates of all points in the description of the contour $\widehat{\mathbf{w}}^{\boldsymbol{e}}$. As a result, we obtain a description $\mathbf{w} = \{\mathbf{w_1}, \mathbf{w_2}, \ldots, \mathbf{w}_{\boldsymbol{k}}\}$ of the object $\mathbf{O}$ with a contour length S. The presented method for obtaining the descriptions of $\mathbf{w} \in K(\mathbf{E}, \mathbf{G})$ reflects, to a certain extent, the model for generating the objects $\mathbf{O}$ by scanning the reference (template) contour $\mathbf{w}^{\boldsymbol{e}}$ chaotically located on the plane XOY and by vectorizing the obtained images. We will then conduct a uniform interpolation of the template $\mathbf{E}$ and the object $\mathbf{O}$ with constant steps $\Delta S^e = S^e/n$ and $\Delta S = S/n$ respectively. As a result, we obtain the descriptions of the contours $\dot{\mathbf{w}}^{\boldsymbol{e}} = \{\dot{\mathbf{w}}_{\mathbf{1}}^{\boldsymbol{e}}, \dot{\mathbf{w}}_{\mathbf{2}}^{\boldsymbol{e}}, \ldots, \dot{\mathbf{w}}_{\boldsymbol{n}}^{\boldsymbol{e}}\}$ of the template $\mathbf{E}$ and $\dot{\mathbf{w}} = \{\dot{\mathbf{w}}_{\mathbf{1}}, \dot{\mathbf{w}}_2, \ldots, \dot{\mathbf{w}}_{\boldsymbol{n}}\}$ of the object $\mathbf{O}$ with the same number

of points. Mathematically, the matching of the contours is given by formula (22.5) and, in accordance with the least squares method; the probability of likelihood reaches its highest value when the value $\varepsilon(\mathbf{A}, \mathbf{\Delta w})$ is minimal, where

$$\varepsilon(\mathbf{A}, \mathbf{\Delta w}) = \frac{1}{n} \cdot \sum_{i=1}^{n} \|\dot{\mathbf{w}}_{\boldsymbol{i}} - \hat{\mathbf{w}}_{\boldsymbol{i}}^{\boldsymbol{e}}\|^2 = \frac{1}{n} \cdot \sum_{i=1}^{n} \|\dot{\mathbf{w}}_{\boldsymbol{i}} - \mathbf{A} \cdot \dot{\mathbf{w}}_{\boldsymbol{i}}^{\boldsymbol{e}} - \mathbf{\Delta w}^{\boldsymbol{e}}\|^2 \tag{22.6}$$

The minimum value ε_m of the function (22.6) indirectly determines the value of the distribution density for the contour $\mathbf{w}$. Thus, if we assume that the probabilities of the classes are equal, then the Bayesian decision rule with the minimum error level chooses the class from among the set of standards for which the value ε_m is the smallest. Hence, the recognition problem has been reduced to an optimization problem of the form

$$\varepsilon_m = \min_{\mathbf{A}, \Delta \mathbf{w}^{\boldsymbol{e}}} \varepsilon(\mathbf{A}, \Delta \mathbf{w}^{\boldsymbol{e}}). \tag{22.7}$$

Let us consider obtaining invariant closeness estimates for various groups of affine transformations based on interpolated descriptions of the object and template contours. Taking the descriptions of the object and the template as a sample of the volume n, we determine such numerical characteristics as the initial moments of the first order $\mathrm{M}\mathbf{w}$ (average $\bar{\mathbf{w}}$), as well as dispersions and correlation moments, of the types $D\mathbf{x}$ and $\mathrm{cov}(\mathbf{x}, \mathbf{y})$ as follows:

$$\mathrm{M}\mathbf{w} = \begin{pmatrix} \mathrm{M}\mathbf{x} \\ \mathrm{M}\mathbf{y} \end{pmatrix} = \bar{\mathbf{w}} = \begin{pmatrix} \bar{x} \\ \bar{y} \end{pmatrix} = \frac{1}{n} \sum_{i=1}^{n} \dot{\mathbf{w}}_{\boldsymbol{i}}$$

$$= \left(\frac{1}{n} \sum_{i=1}^{n} \dot{x}_i, \frac{1}{n} \sum_{i=1}^{n} \dot{y}_i \right)^T,$$

$$\mathrm{M}\mathbf{xy} = \overline{\mathbf{xy}} = \frac{1}{n} \sum_{i=1}^{n} \dot{x}_i \cdot \dot{y}_i, \quad D\mathbf{x} = \mathrm{M}\mathbf{x}^{\mathbf{2}} - (\mathrm{M}\mathbf{x})^2,$$

$$D\mathbf{w} = D\mathbf{x} + D\mathbf{y},$$

$$\mathrm{cov}(\mathbf{x}, \mathbf{y}) = \frac{1}{n} \sum_{i=1}^{n} (\dot{x}_i - \bar{x}) \cdot (\dot{y}_i - \bar{y}) = \mathrm{M}\,\mathbf{xy} - \mathrm{M}\,\mathbf{x} \cdot \mathrm{M}\,\mathbf{y} = \overline{\mathbf{xy}} - \bar{x} \cdot \bar{y}.$$

Denote $\mathbf{v} = (\xi, \eta)^T = \mathbf{w} - \overline{\mathbf{w}}$, $\mathbf{v}^{\boldsymbol{e}} = \mathbf{w}^{\boldsymbol{e}} - \overline{\mathbf{w}}^{\boldsymbol{e}}$, $R^2 = Cs^2 + Sn^2$, where

$$Sn = \operatorname{cov}(\mathbf{x}^{\boldsymbol{e}}, \mathbf{y}) - \operatorname{cov}(\mathbf{y}^{\boldsymbol{e}}, \mathbf{x}), \quad Cs = \operatorname{cov}(\mathbf{x}^{\boldsymbol{e}}, \mathbf{x}) + \operatorname{cov}(\mathbf{y}^{\boldsymbol{e}}, \mathbf{y}). \quad (22.8)$$

22.4.1.1 *The method for obtaining a similarity estimate invariant to a group of orthogonal transformations*

First, we take as an affine transformation a group of orthogonal transformations with respect to which we obtain an invariant estimate of the closeness of the contour descriptions $\varepsilon^t(\alpha, \boldsymbol{\Delta}\mathbf{w}^{\boldsymbol{e}})$. In this case $S^{\mathrm{e}} = S$, $\Delta S^e = \Delta S$ and

$$\mathbf{A} = \begin{pmatrix} \cos\alpha & -\sin\alpha \\ \sin\alpha & \cos\alpha \end{pmatrix}. \quad (22.9)$$

By opening the brackets (22.6) and carrying out simple transformations we get the following expression to calculate $\varepsilon^t(\alpha, \boldsymbol{\Delta}\mathbf{w}^{\boldsymbol{e}})$:

$$\begin{aligned} \varepsilon^t(\alpha, \boldsymbol{\Delta}\mathbf{w}^{\boldsymbol{e}}) = \mathrm{M}\mathbf{w}^2 + \mathrm{M}(\mathbf{w}^{\boldsymbol{e}})^2 - 2\mathrm{M}[\mathbf{w}^T \mathbf{A}\mathbf{w}^{\boldsymbol{e}}] + \\ + 2\,(\Delta\mathbf{w}^{\boldsymbol{e}})^T\, \mathbf{A}\overline{\mathbf{w}}^{\boldsymbol{e}} - 2\overline{\mathbf{w}}^T \Delta\mathbf{w}^{\boldsymbol{e}} + (\Delta\mathbf{w}^e)^T\, \Delta\mathbf{w}^{\boldsymbol{e}}. \end{aligned} \quad (22.10)$$

The minimum ε^t_m and optimal values of the parameters can now be obtained from the solution of problem (22.7).

Theorem 22.1. *If $R \neq 0$, then the extremum points $(\tilde{\alpha}, \boldsymbol{\Delta}\tilde{\mathbf{w}}^{\boldsymbol{e}})$ of the function $\varepsilon^t(\alpha, \boldsymbol{\Delta}\mathbf{w}^{\boldsymbol{e}})$, are found from the formulas*

$$\operatorname{tg}\alpha = \frac{\operatorname{cov}(\mathbf{x}^e, \mathbf{y}) - \operatorname{cov}(\mathbf{y}^e, \mathbf{x})}{\operatorname{cov}(\mathbf{x}^e, \mathbf{x}) + \operatorname{cov}(\mathbf{y}^e, \mathbf{y})} = \frac{Sn}{Cs}, \quad (22.11)$$

$$\boldsymbol{\Delta}\mathbf{w}^{\boldsymbol{e}} = \overline{\mathbf{w}} - \mathbf{A} \cdot \overline{\mathbf{w}}^{\boldsymbol{e}}. \quad (22.12)$$

Theorem 22.2. *The minimum of the function $\varepsilon^t(\alpha, \boldsymbol{\Delta}\mathbf{w}^{\boldsymbol{e}})$ is reached at the extremum point with the value of the parameter α, for which the conditions of the following system are met*

$$\begin{cases} \operatorname{sign}(\cos\alpha) = \operatorname{sign} Cs = \operatorname{sign}[\operatorname{cov}(\mathbf{x}^{\boldsymbol{e}}, \mathbf{x}) + \operatorname{cov}(\mathbf{y}^{\boldsymbol{e}}, \mathbf{y})], \\ \operatorname{sign}(\sin\alpha) = \operatorname{sign} Sn = \operatorname{sign}[\operatorname{cov}(\mathbf{y}^{\boldsymbol{e}}, \mathbf{x}) - \operatorname{cov}(\mathbf{x}^{\boldsymbol{e}}, \mathbf{y})]. \end{cases} \quad (22.13)$$

Theorem 22.3. *The minimum* ε_m^t *of the function* $\varepsilon^t(\alpha, \mathbf{\Delta w}^{\boldsymbol{e}})$ *can be calculated using the following formula:*

$$\varepsilon_m^t = D\mathbf{w}^{\boldsymbol{e}} + D\mathbf{w} - 2 \cdot R. \tag{22.14}$$

Theorem 22.4. *If the descriptions of the template and the object satisfy the condition* $R = 0$*, then at any point of the helix* (VL) *specified by the system of equations* (22.12) *the smallest value of the function* $\varepsilon^t(\alpha, \mathbf{\Delta w}^{\boldsymbol{e}})$ *equal to* $\tilde{\varepsilon}^t = D\mathbf{w}^{\boldsymbol{e}} + D\mathbf{w}$ *is reached.*

The computational complexity of the CECM invariant to orthogonal transformations is determined by the following theorem.

Theorem 22.5. *If the descriptions of the template* $\mathbf{S}$ *and the object* $\mathbf{O}$ *are given by the sequences of uniformly interpolated points, respectively, then the computational complexity of obtaining an estimate of closeness between them* ε_m^t *is proportional to the number of points* $O(n)$ *and is numerically equal to*

$$C^t = (6 \cdot n + 13) \cdot t_y + 8 \cdot n \cdot t_c + t_R, \tag{22.15}$$

where t_c, t_y *and* t_R *— is the speed of addition/subtraction, multiplication/division, and square rooting operations, respectively.*

Corollary 22.1. *The time complexity of calculating the similarity estimates for a group of* L *templates is*

$$C(L) = [(2\,n + 3.5) \cdot (2\,L + 1) - 0.5] \cdot t_y + [(4\,n + 1) \cdot (L + 1) - 2)] \cdot t_c + L \cdot t_R.$$

22.4.1.2 *The method for obtaining a similarity estimate invariant to the group of orthogonal transformations and scaling*

In this case, the transformation of the coordinates of the points corresponding to the matching of the object and the template contours will be written as: $\mathbf{w}^{\boldsymbol{e}} = k_m \cdot \mathbf{A} \cdot \mathbf{w}^{\boldsymbol{e}} + \mathbf{\Delta w}^{\boldsymbol{e}}$, where k_m is the scaling coefficient, and α, $\mathbf{\Delta w}^{\boldsymbol{e}}$, as before, are the orthogonal transformation parameters. By substituting $\mathbf{w}^{\boldsymbol{e}}$ in (22.6) and opening the brackets,

we calculate the value of the residual error $\varepsilon^{\text{ts}}(\alpha,\, \mathbf{\Delta w}^{\boldsymbol{e}},\, k_m)$ in the form

$$\begin{aligned}\varepsilon^{\text{ts}}(\alpha,\, \mathbf{\Delta w}^{\boldsymbol{e}}, k_m) = \text{M}\mathbf{w}^2 + k_m^2\text{M}(\mathbf{w}^{\boldsymbol{e}})^2 - 2k_m\text{M}[\mathbf{w}^T\mathbf{A}\mathbf{w}^{\boldsymbol{e}}]+\\ + 2k_m(\Delta\mathbf{w}^{\boldsymbol{e}})^T\mathbf{A}\bar{\mathbf{w}}^{\boldsymbol{e}} - 2\bar{\mathbf{w}}^T\Delta\mathbf{w}^{\boldsymbol{e}} + (\Delta\mathbf{w}^{\boldsymbol{e}})^T\Delta\mathbf{w}^{\boldsymbol{e}}.\end{aligned} \tag{22.16}$$

The minimum $\varepsilon_m^{\text{ts}}$ of the function $\varepsilon^{\text{ts}}(\alpha,\, \mathbf{\Delta w}^{\boldsymbol{e}},\, k_m)$ and of the values of the optimal transformation parameters are found by solving the problem

$$\varepsilon_m^{\text{ts}} = \min_{\alpha,\, \mathbf{\Delta w}^{\boldsymbol{e}},\, k_m} \varepsilon^{\text{ts}}(\alpha,\, \mathbf{\Delta w}^{\boldsymbol{e}},\, k_m).$$

To find the value $\varepsilon_m^{\text{ts}}$, it is also necessary to first determine the points at which the extreme of the function $\varepsilon^{\text{ts}}(\alpha,\, \mathbf{\Delta w}^{\boldsymbol{e}},\, k_m)$ can exist.

Theorem 22.6. *If the descriptions of the template and the object satisfy the condition* $R \neq 0$, *then the components of the extremum points of the function* $\varepsilon^{ts}(\alpha,\, \mathbf{\Delta w}^{\boldsymbol{e}},\, k_m)$ *are determined on the basis of formulas* (22.11) *and*

$$k_m = \frac{Cs\cdot\cos\alpha + Sn\cdot\sin\alpha}{D\mathbf{w}^{\boldsymbol{e}}} = \frac{R}{D\mathbf{w}^{\boldsymbol{e}}}, \tag{22.17}$$

$$\mathbf{\Delta w}^{\boldsymbol{e}} = \bar{\mathbf{w}} - k_m\cdot\mathbf{A}\cdot\bar{\mathbf{w}}^{\boldsymbol{e}}. \tag{22.18}$$

Theorem 22.7. *The minimum of the function* $\varepsilon^{ts}(\alpha,\, \mathbf{\Delta w}^{\boldsymbol{e}},\, k_m)$ *is reached at both extremum points defined by two parameter* α *values in accordance with formula* (22.11).

Theorem 22.8. *The minimum value* ε_m^{ts} *of the function* $\varepsilon^{ts}(\alpha,\, \mathbf{\Delta w}^{\boldsymbol{e}}, k_m)$ *with* $R \neq 0$ *can be calculated using the following formula:*

$$\varepsilon_m^{ts} = D\mathbf{w} - R^2/D\mathbf{w}^{\boldsymbol{e}}. \tag{22.19}$$

The method for obtaining an estimate of contour closeness based on formula (22.19) *will be referred to as the* ***correlation-extreme contour method***, ***which is invariant with respect to orthogonal transformations and scaling*** *(CECM/OTS)* [2].

Theorem 22.9. *If the descriptions of a template and an object satisfy the condition* $Sn = Cs = R = 0$, *then at any point of the helix given by the system of equations* (22.18), *the smallest value* $\tilde{\varepsilon}_m^{ts}$ *of the function* $\varepsilon^{ts}(\alpha,\, \mathbf{\Delta w}^{\boldsymbol{e}}, k_m)$ *is reached, and this value is* $\varepsilon^{ts} = \tilde{D}\mathbf{w}$.

Theorem 22.10. *If the descriptions of the template* **S** *and the object* **O** *are given by the sequences* n *of uniformly interpolated points, respectively, then the computational complexity of obtaining an estimate of closeness between them* ε_m^{ts} *is proportional to the number of points* $O(n)$ *and is numerically equal to*

$$C^{ts} = (6\,n + 13) \cdot t_y + (8n - 1) \cdot t_c. \tag{22.20}$$

Corollary 22.2. *The time complexity of calculating type similarity estimates* ε_m^{ts} *for a group of* L *templates is*

$$C(L) = [(4\,n + 6) \cdot (L + 0.5) + 4] \cdot t_y + [(4\,n + 1) \cdot (L + 1) + 1)] \cdot t_c. \tag{22.21}$$

22.4.1.3 *The method for obtaining a similarity estimate invariant to the group of affine transformations*

The group **G** of affine transformations on the plane is one of the primitive Lie groups. With the help of his group, the transformation of the template description that corresponds to the matching of the object and the template contours can be specified by the following equation:

$$\hat{\mathbf{w}}^e = \mathbf{G} \cdot \mathbf{w}^e = \mathbf{A} \cdot \mathbf{w}^e + \Delta\mathbf{w}^e, \cdot\mathbf{A} = \begin{pmatrix} a & b \\ c & d \end{pmatrix}, \quad \det \mathbf{A} > 0. \tag{22.22}$$

It is clear that interpolation results before and after applying the affine transformation will differ $(\mathbf{G} \cdot \mathbf{w}^{\boldsymbol{e}})_{\boldsymbol{i}} \neq \mathbf{G} \cdot \mathbf{w}_{\boldsymbol{i}}^{\boldsymbol{e}}$. Therefore, to find the optimal solution

$$\varepsilon_m^{at} = \min_{\mathbf{G}} \varepsilon^{at}(\mathbf{G}) = \min_{\mathbf{G}} \frac{1}{n} \cdot \sum_{i=1}^{n} \left\| \mathbf{w}_{\boldsymbol{i}} - (\mathbf{G} \cdot \mathbf{w}^{\boldsymbol{e}})_{\boldsymbol{i}} \right\|^2, \tag{22.23}$$

the methods used in orthogonal transformations cannot be applied.

The computational complexity of an algorithm for obtaining the similarity estimate $\varepsilon_m^{\text{at}}$ should not depend on a specific affine transformation [2]. To ensure this, we assume that the template lies in the plane XOY of a three-dimensional space, and the group of affine transformations will be represented by a sequence of the following operations: rotation around the axis OX at an angle β (operation

$\mathbf{\Psi}_\beta$), then rotation around the axis OY' at an angle γ (operation $\mathbf{\Psi}_\gamma$). Let us project the resulting figure on the plane XOY (operation $\mathbf{\Psi}_{pr}$). We get the figure $\mathbf{E}_{pr}$ defined by a sequence of points with coordinates (x^e_{pr}, y^e_{pr}) in accordance with the formula

$$\begin{cases} x^e_{pr} = x^e \cos\gamma + y^{\mathrm{e}} \sin\beta \cdot \sin\gamma, \\ y^e_{pr} = y^e \cos\beta. \end{cases} \tag{22.24}$$

Next, we rotate the figure $\mathbf{E}_{pr}$ at an angle θ (operation Ψ_θ) and perform the scaling with the coefficient k_m (operation Ψ_k). As a result, we obtain the figure E that is an image of the template $\mathbf{E}$ with the mapping $\widehat{\Psi} = \Psi_k(\Psi_\theta(\Psi_{pr}(\Psi_\gamma(\Psi_\beta))))$. On the plane XOY, the template $\widehat{\mathbf{E}}$ is defined by the points $\hat{\mathbf{w}}^{\boldsymbol{e}}_{\boldsymbol{i}}$, $i = 1, 2, \ldots, n$, calculated according to the formula

$$\begin{cases} \hat{x}^e = k_m \cdot \cos\theta \cdot x^e_{pr} + k_m \cdot \sin\theta \cdot y^e_{pr}, \\ \hat{y}^e = -k_m \cdot \sin\theta \cdot x^e_{pr} + k_m \cdot \cos\theta \cdot y^e_{pr}. \end{cases} \tag{22.25}$$

Now, it is easy to see that to find the similarity estimate, one can apply decomposition of operations, which consists of obtaining a modified description of the template based on formula (22.24) and then matching the obtained template contour with the object contour. It is easy to guess that the task of matching can be solved by determining the similarity of flat forms of different scales, while the task of changing the description, that is, the form of the template, can be solved by using the grid method.

Teopema 22.11 If $\cos\beta > 0$ *and* $\cos\gamma > 0$, *then the transformation* $\widehat{\Psi}$ *is an affine mapping of the first kind, and vice versa, for any affine mapping of the first kind there exists a transformation* $\widehat{\Psi}$ *that implements it.*

22.4.2 *Parametric optimization methods for calculating the similarity estimates*

It follows from the analysis of the results obtained that the speed of the methods and the accuracy of calculating the closeness estimate itself can be increased by using the original (chain-node) description

of the forms instead of the interpolated representation. Specifying the template n by means of evenly spaced points implies interpolation of the template contour $\mathbf{E}$ with a step $\Delta S^e = S^e/n$. It is obvious that all the interpolated points of the contour located between the nodes with the numbers j and $(j+1)$ will be calculated by the formula

$$\mathbf{w}^e_{j,\tau} = \mathbf{w}^e_j + (\tau \cdot \Delta S - \Delta P_j) \cdot (\mathbf{w}^e_{j+1} - \mathbf{w}^e_j)/S^e_j, \tag{22.26}$$

where S_j is the length of the jth edge, ΔP^e_j $(0 \le \Delta P^e_j < \Delta S^e)$ is the remainder obtained after interpolation with a given step ΔS^e of the previous contour segment, and τ is the serial number of the point. If we substitute the values from (22.26) into the calculation formula $\mathrm{M}\mathbf{x^e}$ and go to the limit at $\Delta S^e \to 0$ in the resulting formula, we will have

$$\mathrm{M}\mathbf{x^e} = \bar{x}^e = \frac{1}{2\,S^e} \sum_{j=1}^{m-1} (x^e_j + x^e_{j+1}) \cdot S^e_j. \tag{22.27}$$

In a similar manner, we can obtain a formula for calculating all the six sample moments of the second order:

$$\mathrm{M}\mathbf{x}^2 = \frac{1}{3S} \cdot \sum_{j=1}^{k-1} \left[x^2_j + x^2_{j+1} + x_j\, x_{j+1}\right] \cdot S_j. \tag{22.28}$$

In order to use formula (22.26) for calculating mixed moments of the second order, it is necessary to form auxiliary descriptions of the template and the object. To do this, we virtually reduce the length of the template contour to the length of the object contour and map the position of the obtained template points on the edges of the object. No more than $(m-2)$ additional nodes will appear in the object description. A similar operation will be performed with the description of the template, but the number of nodes inserted will be $(k-2)$. Auxiliary descriptions of the object and the template do not improve the quality of the descriptions themselves but have the same number of points $l = (k+m-2)$, and the distances between the nodes with the same numbers are proportional to $\tilde{S}_j/\tilde{S}^e_j = (S/S^e)$. Hence, the description of the object will be $w = \{(\tilde{x}_1, \tilde{y}_1), (\tilde{x}_2, \tilde{y}_2), \ldots, (\tilde{x}_l, \tilde{y}_l)\}$, and that of the template, $w^e = \{(\tilde{x}^e_1, \tilde{y}^e_1), (\tilde{x}^e_2, \tilde{y}^e_2), \ldots, (\tilde{x}^e_l, \tilde{y}^e_l)\}$. As

a result, we get [2]

$$\mathrm{M}\mathbf{x}^{\boldsymbol{e}}\mathbf{y} = \frac{1}{6\,S^e}\sum_{j=1}^{l-1}\left[2\,(\tilde{x}_j^e\,\tilde{y}_j + \tilde{x}_{j+1}^e\,\tilde{y}_{j+1}) + \tilde{x}_j^e\,\tilde{y}_{j+1} + \tilde{x}_{j+1}^e\,\tilde{y}_j\right]\cdot\tilde{S}_j^e. \tag{22.29}$$

22.4.3 *The method for finding the symmetry axes of figures*

The use of the axial symmetry of the template almost doubles the speed when calculating the closeness estimate. In addition, axial symmetry can be used to structure the recognition procedure by dividing the entire set of templates into two classes in advance. Thus, the recognition problem can be solved using a smaller set of templates.

Suppose there is some contour $\mathbf{E}$ defined by a sequence of nodes $\mathbf{A_i}(x_i^e,\, y_i^e)$, $i = 1, 2, \ldots, m$. Let the line γ be the symmetry axis of the figure. On the plane XOY, any equation of the line can be written as follows:

$$\gamma:\ \ x\cdot\cos\phi + y\cdot\sin\phi + c = 0. \tag{22.30}$$

We find the image of the contour $\mathbf{E}$ with respect to the line γ. As a result, we obtain some new contour $\bar{\mathbf{E}}$, which is defined by a sequence of points $\mathbf{B_i}(x_i,\, y_i)$, $i = 1, 2, \ldots, m$. If the line γ is the axis of symmetry, then it must pass through the midpoint of the figure $(\bar{x}^e, \bar{y}^e)$. From here, we find the parameter c in the equation of the line γ: $c = -\bar{x}^e\cdot\cos\gamma - \bar{y}^e\cdot\sin\gamma$. Minimizing the closeness estimate $\varepsilon^s(\phi,\, c)$ of the contour descriptions $\mathbf{E}$ and $\bar{\mathbf{E}}$, $\varepsilon_m^s = \min\limits_{\phi,c}\varepsilon^s(\phi,\, c)$ allows us to find the remaining parameters of the line [2]

$$tg2\phi = \frac{Sns}{Css},\quad \varepsilon_m^s = 2\cdot(D\mathbf{w}^{\boldsymbol{e}} - \sqrt{Css^2 + Sns^2}), \tag{22.31}$$

where $Css = \mathrm{cov}(\mathbf{x}^{\boldsymbol{e}},\, \widehat{\mathbf{x}}^{\boldsymbol{e}}) - \mathrm{cov}(\mathbf{y}^{\boldsymbol{e}},\, \widehat{\mathbf{y}}^{\boldsymbol{e}})$, $Sns = \mathrm{cov}(\mathbf{y}^{\boldsymbol{e}},\, \widehat{\mathbf{x}}^{\boldsymbol{e}}) + \mathrm{cov}(\mathbf{x}^{\boldsymbol{e}},\, \widehat{y}^{\boldsymbol{e}})$.

Note that the value ε_m^s found can serve as an indicator of the figure's symmetry. Also note that any description of the contour can give the position of one symmetry axis only. To obtain all the axes of symmetry, it is necessary to find all the corresponding cyclic contour descriptions.

22.4.4 *Methods for obtaining consistent descriptions*

Template description is one of the multitude of ways to define the contour that only differs by the point which is taken as the initial one, that is, the point that generates the equivalence class $K(\mathbf{C_E}, \mathbf{G})$, where $\mathbf{C_E}$ is the contour represented by the template $\mathbf{E}$. The entire set of contour representations can be described parametrically using the contour displacement by the value s relative to the original (basic) contour definition. Therefore, to ensure invariance with respect to the choice of the initial point of the object description, it is necessary to solve the following optimization problem:

$$\varepsilon_{\min} = \min_{s\in[0,S]} \varepsilon_m(s), \tag{22.32}$$

where S is the length of the object contour. In what follows, the descriptions of the template and the object for which condition (22.32) is satisfied will be called consistent, and the task of obtaining them reduces to the task of determining the location of the global minimum for the closeness estimate $\varepsilon_m(s)$. The analysis of the functions $\varepsilon_m(s)$ shows that the task of finding the global minimum should be divided into two tasks: the task of localization and the task of refining its location [2].

22.4.4.1 *Relative displacements method*

The essence of this method is to find the minimum value of the closeness estimate $\varepsilon_m(\mathbf{E}, \mathbf{B}, s)$ (22.32) for any template $\mathbf{E}$ to the reference template $\mathbf{B}$, for which the Pythagorean triangle is used (the minimum number of points in the description and the absence of any symmetry). The value of the displacement $s^*_{\mathbf{E}}$ for which the minimum value is reached is determined at the stage of template formation and therefore it has no influence on the speed of the recognition algorithm. Similarly, the displacement s^* is found that matches the description of the object $\mathbf{O}$ with the reference template $\mathbf{B}$. Then the localization of the region of the function global minimum consists in moving the initial point of the object description by an amount $s_o = s^* - s^*_{\mathbf{E}}$ in the direction of the contour traversal.

22.4.4.2 *Methods for refining the global minimum location*

The functions $\varepsilon_m(\mathbf{E}, \mathbf{O}, s)$ have a number of properties that make it possible to develop and implement high-speed methods for refining the location of the global minimum.

Theorem 22.12. *The function $\varepsilon_m(\mathbf{E}, \mathbf{O}, s)$ is a polynomial of the sixth degree in the parameter s.*

Theorem 22.13. *If the template and the object belong to the same equivalence class $K(\mathbf{C_E}, \mathbf{G})$, then the function $\varepsilon_m(\mathbf{E}, \mathbf{O}, s)$ is even with respect to the location of the global minimum $s_o = s^* - s^*_{\mathbf{E}}$.*

Parabolic method. These theorems imply a well-founded assumption that in the global minimum region, the function $\varepsilon_m(\mathbf{E}, \mathbf{O}, s)$ in the chosen interval can be approximated by a parabola. Obviously, to obtain the parabola equation, it is necessary to specify three points, that is, to calculate the three values of the function $\varepsilon_m(\mathbf{E}, \mathbf{O}, s)$. Then the vertex of the parabola will give the refined location of the global minimum, and the calculated value of the function $\varepsilon_m(\mathbf{E}, \mathbf{O}, s)$ at this point will be an approximate value of the similarity estimate $\varepsilon_{\min}(\mathbf{E}, \mathbf{O})$. The described procedure for refining the location of the global minimum is called the **parabolic method**.

22.4.5 *Structural optimization methods*

With a great variety of objects in the image (the number of object classes amounting to hundreds, thousands, and more), the requirement to reduce the time complexity of the CECM-based recognition algorithm continues to be relevant. The initial scheme of any recognition algorithm based on the method of comparison with templates is aimed at calculating all similarity estimates.

Let us consider the approach involving the construction of a classifier with a tree structure for some group of templates. To solve the problem of constructing a classifier the first step should be to form a set of features. Since we have to calculate an estimate of the similarity with the basic template $\mathbf{B}$, it is appropriate to take this characteristic as the first feature c_1 of the description being generated. Thus, $c_1 = \varepsilon_{\min}(\mathbf{B}, \mathbf{Z})$. Practically without any computational expense, we can get the second feature c_2. It is an estimate of the similarity of the

basic template to the object $\overleftarrow{\mathbf{Z}}$ obtained as a result of a change in the contour traversal. Thus, $c_2 = \varepsilon_{\min}(\mathbf{B}, \overleftarrow{\mathbf{Z}})$. Next, in order to build decision rules for partitioning (if necessary) into smaller subsets, new features are formed for each subset from the estimates of closeness with resident templates. To speed up the procedure, these closeness estimates are calculated only from the coarse matching descriptions. As a result, one to several templates will be left on each classifier leaf [2].

22.5 Intellectual Technology of Vectorization of Graphic Documents

The method for determining the similarity of flat forms of different scales has turned out to be the most popular method for solving practical problems. The low computational complexity of obtaining similarity estimates makes it possible to use this method, for example, to recognize discrete objects with a large number of templates (more than 1000).

22.5.1 *Recognition of symbols and inscriptions*

To solve the problem of automating the input of information when creating digital maps, correlation extreme contour methods were used to recognize symbols and inscriptions on maps for their subsequent identification. We used the method based on finding closeness estimate $\varepsilon_m^{\text{ts}}$, which is invariant with respect to orthogonal transformations and scaling. Recognition was performed for topographic fonts and thematic sets of conventional symbols. To speed up the recognition of individual characters, classifiers with a hierarchical structure were built for the 11 most commonly used fonts. To assemble the inscriptions, we used the values of the parameters obtained as a result of CECM recognition, including the ASII code, location, orientation, size, register, and font. Recognition of sets of conventional signs and inscriptions has been successfully applied in the technologies for digitizing topographic maps and plans, as well as nautical charts and plans.

22.5.2 *IFc intelligent format*

A characteristic feature of graphic documents, including text documents, is the presence of a large number of randomly located objects of a certain size that have a rigid contour description, which are commonly referred to as discrete objects. Some examples of discrete objects are symbols and conventional signs of the domain of the graphic document under consideration. However, despite the large number of discrete objects in raster documents, their variety is significantly limited. Therefore, the same reference (template) description can be used to represent any discrete object from this class [13].

Traditional compression is replaced by the recognition of objects when fixed-length records are formed that include the location, rotation, and scaling of a discrete object relative to the template. In total, the length of the record containing the reference to the template number does not exceed 8 bytes. Recognition is carried out in self-learning mode based on the closeness estimate $\varepsilon_m^{\text{ts}}$. The list of templates is replenished by adding unidentified objects. Small-sized objects are encoded using the bitmask method. Large-format objects in this format are encoded either by differences, or increments, or pointwise, depending on the result obtained. In order to better represent discrete objects, a second, additional closeness estimate can be used in recognition based on the calculation of the Hausdorff distance

$$d_H = \max\left\{\sup_{\mathbf{w}\in\mathbf{0}}\inf_{\tilde{\mathbf{w}}\in\mathbf{0}}\|\mathbf{w}-\tilde{\mathbf{w}}\|, \sup_{\tilde{\mathbf{w}}\in\tilde{\mathbf{O}}}\inf_{\mathbf{w}\in\mathbf{O}}\|\mathbf{w}-\tilde{\mathbf{w}}\|\right\}, \qquad (22.33)$$

where $\tilde{\mathbf{O}}$ is the object restored on the basis of the template with the best closeness estimate. The calculation of this estimate is carried out only with the use of auxiliary descriptions when the template matches the object in the optimal manner.

22.5.3 *Input technology for documents with a weakly formalized description of objects*

For documents with a weakly formalized description of objects, the vectorization technology consists of obtaining two sets of templates

resulting from one-criterion recognition with self-learning and two-criteria recognition with self-learning using additional estimates for the Hausdorff distance [4]. The second set of templates guarantees the quality of recognition and the elimination of errors when creating a digital model. The essence of the IA-technology is as follows [15]. The first set of templates is marked up interactively. Then, in an automatic mode, each template of the second group is assigned an identifier of that template from the first group that gave the best closeness factor during recognition. Further, by displaying all templates with the same identifier from the second set, it will be possible to quickly verify them in an interactive mode.

22.6 Conclusion

There are a number of reasons to develop new methods, algorithms, and technologies or to modify the existing ones. These reasons include, first of all, the lack of the necessary methods or the lack of access to such methods and their low efficiency in solving applied problems in this particular field. That is why, to create digital models of large-format raster documents, we have developed correlation extreme contour recognition methods, general-to-specific methods based on hierarchical structures for 2D and 3D information representation, and methods for decision-making on these structures for solving geometric problems. A new intelligent IFc format was introduced, which offers a compression ratio that is several times higher than that of the DjVu algorithm [13, 16]. Intelligent information technologies and vectorization software for raster large-format graphic documents with a weakly formalized description of symbols and inscriptions (topographic maps, city plans, etc.) were developed on the basis of recognition methods with self-learning and computational geometry methods. The developed algorithms and their software implementation for the synthesis of basis functions, coding, adaptive compression, computational geometry, and CECM recognition are distinctive in their novelty, low time and space complexity; the coding and adaptive compression algorithms offer a high compression ratio.

Experimental testing of the new technology was performed using actual graphic documents (hydrographic survey sheets) and showed a significant increase in performance (10–15 times depending on the

quality of documents) compared with traditional interactive technologies for creating digital graphic documents [15].

And, finally, experiments using the basis's **U**, **G**, **K** and the modified CECM algorithms have demonstrated that they are highly efficient in a new field of research where spatial analysis of hyperspectral images is required.

Acknowledgments

In this chapter, we used the research results obtained by the authors in the framework of a grant project of the Russian Science Foundation (project No. 16-11-00068-P).

References

[1] Lebedev, L.I. (1995) Simple decision rules in problems of pattern recognition and graphic information processing. DisserCat-Electronic Library of Dissertations, Russia, 188 p. (in Russian).

[2] Lebedev, L.I. (2013) *Correlation — Extreme Contour Recognition Methods. Theoretical Foundations: A Study Manual* (Publishing House of the Nizhny Novgorod State University, Russia), 113 p. (in Russian).

[3] Lebedev, L.I. and Vasin, Y.G. (2015) Two-criteria algorithm for recognition of objects in graphic images based on CECM. *Proceedings of 25th International Science Conference, GRAPHICON'2015*, pp. 112–114 (in Russian).

[4] Lebedev, L.I., Shakhlan, A.O., and Vasin, Y.G. (2017) Optimization of computational complexity in a two-criteria recognition algorithm with self-learning. *Proceedings of 3th International Conference "Information Technology and Nanotechnology", ITNT'2017*, pp. 737–740 (in Russian).

[5] Neimark, Y.I., Batalova, Z.S., Vasin, Y.G., and Breido, M.D. (1972) *Pattern Recognition and Medical Diagnosis* (Nauka Press, Moscow), 328 p. (in Russian).

[6] Vasin, Y.G. (1978) Irregular sampling of initial information samples and the problem of electrocardiographic data coding. *Journal of Cybernetics and Computers*, 42, 98–104 (in Russian).

[7] Vasin, Y.G. (1979) "*Well-adapted*" *Bases and Problems of Experimental Information Processing: A Study Manual* (Publishing House of the Gorky State University, Russia), 129 p. (in Russian).

[8] Vasin, Y.G. (1979) Chapter: Current state of the operations research theory. *Optimization of Source Data Description in Dialog Systems for Solving Classification Problems* (Nauka Press, Moscow), pp. 424–446 (in Russian).
[9] Vasin, Y.G. (1984) "Well-adapted" local homogeneous methods for graphic information processing. *Proceedings of the Gorky State University "Automated Processing of Complex Graphic Information"*, Inter-University Collection of Papers, pp. 131–158 (in Russian).
[10] Vasin, Y.G. (1986) Efficiency of various strategies for video information processing on the basis of local homogeneous recurrently-recursive functions. *Proceedings of the Gorky State University "Methods and Tools for Processing Graphic Information"*, Inter-University Collection of Papers, pp. 4–45 (in Russian).
[11] Vasin, Y.G., Krakhnov, A.D., and Utesheva, T.S. (1988) The general-to-specific method in solving spatial problems of discrete geometry. *Proceedings of the Gorky State University "Automated Processing of Complex Graphic Information"*, Inter-University Collection of Papers, pp. 73–83 (in Russian).
[12] Vasin, Y.G., Krakhnov, A.D., and Utesheva, T.S. (2004) Methods of discrete geometry in the problems of complex graphic information processing. *Proceedings 7th International Conference "Pattern Recognition and Image Analysis: New Information Technologies"*, PRIA-7-2004, Vol. 3, pp. 954–956 (in Russian).
[13] Vasin, Y.G. and Lebedev, L.I. (2012) An effective format for representing graphic information. *Journal of Pattern Recognition and Image Analysis. "Advances in Mathematical Theory and Applications"*, 22(2), 393–398.
[14] Vasin, Y.G. and Utesheva, T.S. (2013) Raising the efficiency of technological processes of digitizing and preparation of graphical information for 3d modeling and rendering. *Proceedings of 11th International Conference "Pattern Recognition and Image Analysis: New Information Technologies"*, PRIA-11-2013, Vol. 2, pp. 763–765 (in Russian).
[15] Vasin, Y.G. and Lebedev, L.I. (2016) Intellectualization of information technologies for producing digital graphic documents with weakly formalized description of objects. *Journal of "Pattern Recognition and Image Analysis. Advances in Mathematical Theory and Applications"*, 26(3), 638–642.
[16] http://www.djvu.org/.
[17] https://www.abbyy.com/.
[18] https://win-apps.ru/cuneiform-12-rus/.

© 2025 World Scientific Publishing Company
https://doi.org/10.1142/9789811267215_0023

Chapter 23

Methods for Aerospace Images and Raster Topographic Maps Processing to Create and Update Geospatial Data

Vitaly S. Titov*, Sergey Yu. Miroshnichenko†, and Vladimir S. Panishchev‡

Computer Sciences Department of Southwest State University, Kursk, the Russian Federation

*titov-kstu@rambler.ru

†*oldguy7@rambler.ru*

‡*gskunk@yandex.ru*

This chapter is devoted to aerospace images and raster topographic maps processing and recognition methods application to create and update vector maps and digital elevation models (DEMs), that form geoinformation basis to solve computational problems (for instance, relocation routes multicriterial construction) and visualize terrain.

Methods for objects with different geometric shapes detection in set up by user calibration points on aerospace images are considered. The method provides automatic detection of the object's parameters (for instance, rectangle's position, orientation, and aspect ratio). A method for buildings' and constructions' projection to the basement, providing their true geographic positions definition, and a method for objects' shades detection, designed to calculate their heights over the underlying surface, are described. Experimental investigations of software instruments, created using mentioned above methods, are shown. Vectorization precision and velocity values compared to known analogs are presented.

Methods for isolines detection, height marks, and isolines' signatures recognition on raster topographic maps, that were scanned from paper medium, and their composition to vector and digital elevation models are described. Experimental investigation results, namely, precision and velocity values, of DEM construction in automatic and automated modes are presented.

Keywords: GIS, raster terrain images, vector terrain images, digital surface model, digital terrain model, imagery digitizing, terrain object, map-to-DSM transformation, isoline, elevation sign.

23.1 Introduction

The range of modern geoinformation systems (GIS) and technologies applications is very wide: cadastre and municipal administration, public transport and private transportation, logistics, agriculture and forestry, marine management, environmental monitoring, unmanned vehicle management, emergency prevention and response, planning, and monitoring of special operations. Such a significant coverage of various fields of activity is caused by the fact that GIS uniquely combines the possibilities of integrating a variety of data on a geocoordinate basis, user-friendly 2D and 3D visualization, spatial analysis of information, and support for decision-making. The flip side of GIS widest functionality is the constant need to add and update geospatial information, caused by the necessity to expand the application area and obsolescence of information and, as a consequence, the constant need to actualize it.

The informational basis of GIS, which is necessary both for solving computational problems and for visualization, is

- raster terrain images obtained by aerial and space means of Earth's surface remote sensing,
- vector terrain maps with a standard set of thematic layers which includes vegetation, hydrography, buildings and structures, road network, territorial and administrative boundaries,
- digital surface model (DSM) and digital terrain model (DTM), usually presented as regular elevation matrices.

Raster terrain images are a product of remote sensing data (RSD) primary processing that is aimed at compensating optical system's

aberrations, correcting atmospheric and radiometric effects, rectifing image to obtain a normal parallel projection in the desired coordinate system, and improving visual properties [35].

Vector (digital) maps are created and updated within the secondary RSD processing which consists of two main processes:

- digitizing aimed to construct vector representations of terrain objects,
- recognition performed to subdivide digitized objects into a number of thematic (vector) layers and to determine their semantic properties.

DSM and DTM are also a product of secondary RSD processing and are created to reconstruct elevation models from an image's stereo pair, a set of frames with overlapping visible areas, or a radar image using specialized methods and data processing tools [45]. If aerial or space stereoscopic optical or radar surveys are inaccessible, a good alternative is a DSM construction from raster topographic maps of different scales which have been scanned from paper and contain a relief model in the form of single-height contours (isolines), altitude values, and microrelief signs.

The authors developed methods and created automated software tools to process RSD and raster topographic maps, which compared with existing analogues on the market, allow us to reduce the time consumption to prepare and update the main components of GIS information basis.

23.2 Methods of Remote Sensing Data Processing

Aerial or space imagery digitizing in most cases is performed manually by delineating the boundaries of the desired terrain objects by an operator with the help of interactive software, the most well known of them are GIS Panorama, MapInfo, Microstation, and EasyTrace [10, 26, 27, 32]. Digitizing is an extremely painstaking and time-consuming process, especially when an operator detects building contours in dense urban areas or contours of natural objects, such as vegetation and hydrography that do not have a clear geometric shape and require a significant number of points (usually more than 50) to describe their borders. The precision of the created

vector representations directly depends on an operator's qualification and experience, his ability to analyze the image, and visually detect the boundaries of terrain objects. The precision is negatively affected by operator fatigue, accumulated due to the monotony of performed operations. As a consequence, the digitizing process with the described interactive approach requires significant time costs.

Currently, there are software tools for automated RSD processing designed to reduce time costs for digitizing and provide a workflow acceleration for each operator individually by increasing an automation ratio.

The complex for automated recognition and digitizing [7] has a learning classifier that uses statistical (average color value, standard deviation) and texture characteristics (contrast, energy, and correlation). It is used to detect objects with a uniform brightness or texture, such as vegetation, hydrography, or roads. It is the only software product that has the ability to export to GIS Map and correctly supports Russian coordinate systems: SK-42, SK-95, PZ90.02, PZ90.11.

However, the complex has a small variety of features and classification algorithms that have low robustness and noise immunity and require considerable time to adjust their parameters.

Feature Analyst, Envi FX, and eCognition [2,12,25] software complexes for automated RSD processing that allow wizards to create individual rules and step-by-step sequences to recognize and digitize spatial objects. These products are based on object-oriented detection [21] and are aimed to significantly accelerate digital maps' creation, but, in practice, in spite of declared features their range of application comes down to greenery and hydrography objects digitizing and calculative and estimative problems solution due to the following drawbacks:

- the tools are not intuitive in use, and the time required to tune the toolkit and training it is sometimes much greater than the interactive digitizing of required terrain objects,
- small applicability for digitizing of single-channel images due to the small dimension of the feature space and the impossibility to calculate the chromatic indices [11, 30, 37, 42],
- algorithms of automatic classification have insufficient spatial accuracy for the cartographic products, which is characterized by

deviation of vector representations from the true boundaries of objects in the image,
- there are no tools for automatic shadow detection and projecting the vector representations into the true ground positions of the corresponding terrain objects,
- structure of the image, its properties, and features of terrain objects are not taken into consideration,
- require high operator qualification and knowledge both in cartography and image processing, are more intended for specialists in data classification and machine learning than for cartographers.

The result of automated digitizing is rather a "sketch", which allows you to estimate the number of objects of a certain type in the image, their spatial positions, and the occupied area. Correction of such a map requires to verify and edit the vector representation of each object, which, at least, is not inferior to the interactive digitizing of them from scratch.

ERDAS IMAGINE software contains the region grow tool that in combination with a set of vector cleanup operations provides automated user-controlled objects digitizing [31], however the region grow tool has significant computational complexity (a created polygon appears after at least a couple of seconds after the user action). Object digitizing is separated from the vector cleanup process that is performed for the whole layer causing more time consumption for output precision control compared to an object-to-object approach.

Feature Analyst contains a toolset for assisted extraction that allows user to digitize a building simply by dragging a rectangle that encompasses it [3]. The shortcoming of this tool is that the building must be oriented strictly along the screen axes.

Results of investigations and competitions in RSD recognition demonstrate the perspectivity of using artificial neural networks (ANN), which compared to the described above classic machine learning methods and tools have the following advantages:

- higher spatial accuracy of the created vector layers, less probability of type I and II errors for objects' classification [13],
- higher noise immunity and robustness for image parameters and sensor properties,
- partial use of image structure, objects' features images due to the generalizing properties of ANN.

Modern neural network architectures combined with GPGPU computing allow to find effective solutions of the following space imagery recognition problems: to detect objects of a priori specified one (for instance roads in [18] and vehicles in [19, 34]) or several classes (up to 10 in [9, 20, 23, 24, 39]), to find and count certain kinds of animals [22, 29].

The disadvantages of ANN include the following:

- even higher learning complexity comparing to classic algorithms, a thin line between insufficient training and overtraining [17]
- there is no implementation in complexes for RSD processing — ANNs are accessible only to highly skilled specialists in machine learning in the form of software packages for tensaor and neuro calculations, such as Torch, Cafe, Tenzorflow, and Theano. [5, 38, 40, 41].

Based on the analysis results of the existing software for RSD processing, the authors propose to develop methods for automated digitizing that will be used as a base for the toolkit with an easy-to-use interface designed for cartographers to reduce the number of interactive digitizing routine operations [16]:

The following methods for RSD processing and digitizing are developed:

- method to digitize a single object of rectangular, circular, orthogonal, and arbitrary shape at a marked point of the aerospace image,
- method to digitize natural area sites (vegetation, hydrography, and clouds) marked by a set of reference points,
- method to reproject the vector representations into the true ground positions, detect shadows, and calculate the heights of the corresponding terrain objects.

23.2.1 *Method to digitize a single terrain object of the desired shape*

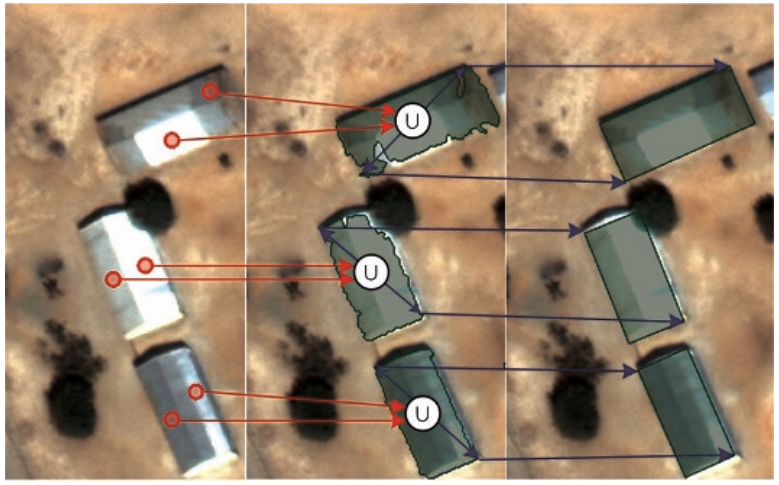
Method to digitize a single terrain object of the desired shape is designed to detect its boundaries and create the corresponding vector representation using initial markup containing one or several reference points specified by the operator. The method reconstructs an object's reference "portrait" describing it with statistical and

textural features, that can be calculated both from monochrome and multichannel images, with the subsequent detection of the image's region satisfying the created "portrait" (Figure 23.1).

Hough transform [1, 15] is the method to provide automatic conversion of selected region to a rectangular (Figure 23.2(a)), circular (Figure 23.2(b)) and orthogonal shape (Figure 23.2(c)). Figure 23.2(b) proves the method's ability to digitize partially detected objects (a half the round object is enough to digitize it correctly). If natural objects are digitized (for instance, vegetation in Figure 23.2(d) or hydrography), the shape of the created vector may have no corrections.

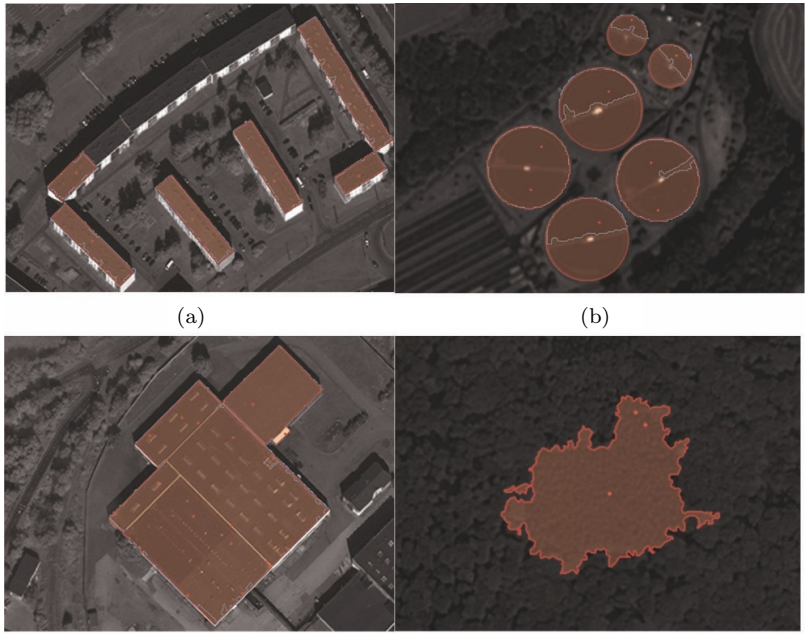
This method has a number of parameters which include the desired shape, sensitivity, a minimum size of the detected object, and a smoothing radius of vector representation for an arbitrary shape.

The "Sensitivity" parameter affects the size of the image region detected using the "portrait" which gives an operator the ability to control the detected region's size and to adjust it to the visible object's borders. Figure 23.3 shows the results of digitizing an orthogonal object for different values of this parameter:



(a) (b) (c)

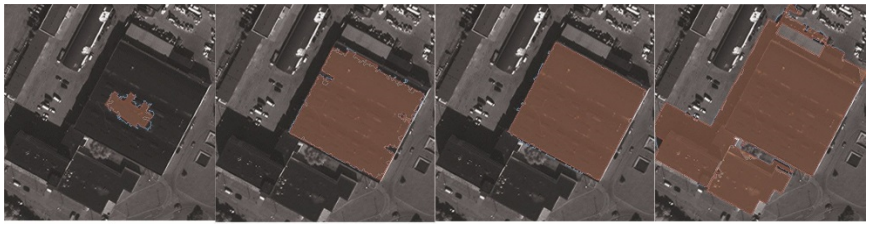
Figure 23.1. Visualization of automated digitizing process: use of reference points (a) to detect the corresponding objects (b) and to transform them to a rectangular shape (c).



(a) (b)

(c) (d)

Figure 23.2. Digitizing the rectangular (a), round (b), orthogonal (c), and arbitrary (d) objects in RSD (border of the detected object without correction of its shape is in blue).



(a) (b) (c) (d)

Figure 23.3. Digitizing of an orthogonal object with different values of the "Sensitivity" parameter.

- Figures 23.3(a) and 23.3(b) represent insufficient values to detect the whole object, as the created vector lies inside the true boundaries,
- the value in Figure 23.3(c) is optimal for the object and allows it to digitize it according to the true boundaries,

- the value in Figure 23.3(d) seems to be redundant as the created vector representation lies outside the true boundaries and includes the shadow of the object and adjacent buildings.

The authors conducted the experiment to compare the precision and speed of the digitizing workflow with the developed method compared to the corresponding indicators for the interactive digitizing toolset of GIS Mapinfo [26]. A test dataset contains a variety of urban areas images of 0.5 m per pixel resolution including multi-apartment residential buildings and industrial zones. The dataset was used to create the reference vector map with 777 rectangular buildings and structures. Two experts were involved in the experiment. Their task was to create the same map with the help of interactive tools and automated software based on the developed method.

The quantitative expression of the precision for the maps created by both experts was calculated relative to the reference map using the following criterions:

(1) root mean square error (RMSE) objects points' positions on the created map which is expressed in pixels and is calculated as a distance to the nearest point of the corresponding reference object,
(2) mean rotational error (MRE), expressed in degrees and calculated as a minimum angle between the lines forming created and corresponding reference objects,
(3) Jaccard index (dimensionless, expressed in percentage) represents the ratio of intersection area of a created object and a corresponding reference one to their unions' area [20].

Within the experiment, experts created four maps (two of them by interactive tools, and the other two by the developed method), mean error values and the speed of digitizing are presented in Table 23.1. Digitizing workflow time consumption decreased by 39.9% and 26.3% for the first and the second expert, respectively. Error values of digital map created by the second expert using the developed method although are slightly greater compared to the interactive method but still are within the range of acceptable values. The first expert gained a more significant time economy but exceeded the maximum RMSE and MRE values that for our experiment are set to 2 pixels (mean error of object's border visual detection in an original image scale

Table 23.1. Average values of digitizing speed and accuracy parameters for maps created by experts.

Expert	Method	Number of objects	Digitizing time, min	Time for an object, sec	RMSE, pixels	MRE, degree	Jaccard index, %
Expert 1	Interactive	775	180	13.94	1.779	0.751	90.93
	Developed	773	**108**	**8.38**	2.251	1.053	88.89
	Difference		–72 (–40%)	–5.56 (–39.9%)	+0.472 (+21%)	+0.302 (+28.7%)	–2.04
Expert 2	Interactive	773	188	14.59	1.567	0.728	91.33
	Developed	770	**138**	**10.75**	1.909	0.925	90.33
	Difference		–50 (–26.8%)	–3.84 (–26.3%)	+0.342 (+17.9%)	+0.197 (+21.3%)	–1

without zooming) and 1 degree (determined by the quantization of histogram used to calculate a gradient angle mode).

23.2.2 *Method to digitize natural area sites*

Method of digitizing natural area sites is designed to form thematic layers of vegetation, hydrography, and clouds. As the method described above this one is similarly based on an a priori set of reference points that forms a "portrait" of digitized objects in a space of statistical and textural features. The particularity of the method is the need to split the set of reference points into subsets of "positive" (representing the objects to be digitized, shown in red squares in Figure 23.4), and "negative" points (objects that are not required to be detected are shown in green rhombs in Figure 23.4). Two subsets allow us to determine the boundaries of the digitized objects class and to adjust the decisive rule in the statistical and textural features space. The textural features are calculated with the MR8 filters set [43], whose responses are further used to obtain rotationally invariant values of fractal dimension and fractal length [6].

Within- and between-class distances in the feature space are used to create the separating hypersurface used for RSD pixel-wise classification with the subsequent morphological correction, local alignment of objects' boundaries according to gradient vectors, and their transformation into a vector form. The computational complexity of the method can be reduced through multiscale image processing with the preliminary constructed three-level pyramid [4]. First, rough

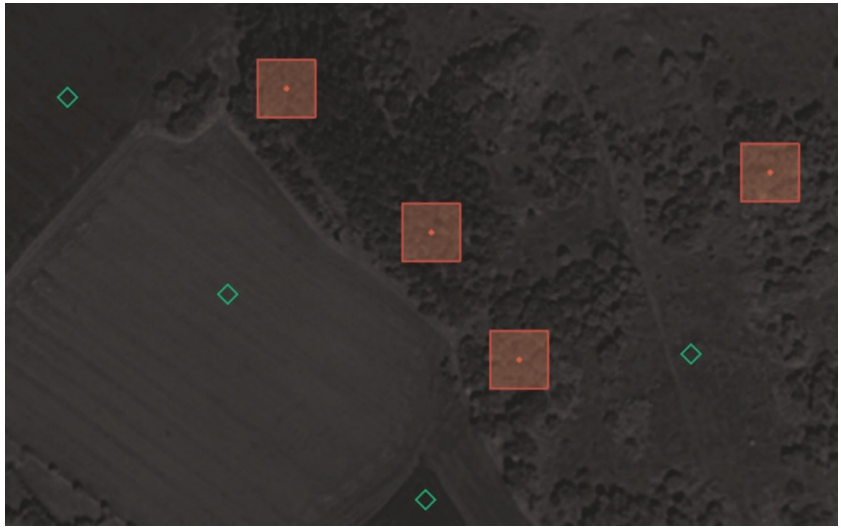


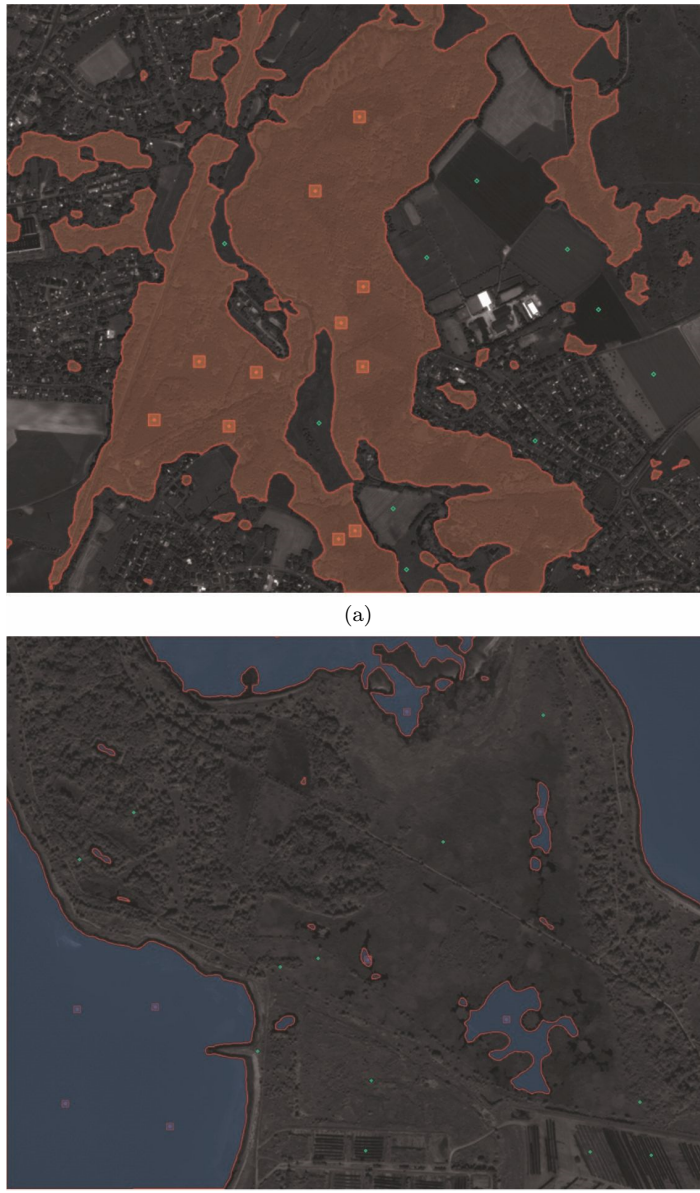
Figure 23.4. "Positive" and "negative" reference points setup to digitize the vegetation.

objects' detection using texture "portrait" is performed on the top level (25% of the original image), whereupon the generated "draft" becomes more detailed on the middle (50%) and lower (100%) levels, respectively. An example of the described method's application to digitize the vegetation is given in Figure 23.5(a), hydrography in Figure 23.5(b).

The described method is used to develop a tool for digitizing the natural area objects, which compared to the analogs implemented in Feature Analyst, Envi FX, and eCognition, does not require an operator to have knowledge of image processing and machine learning and allows us to reduce the time costs for its adjustment by 50%.

23.2.3 *Method to reproject buildings into the true ground positions and calculate their heights*

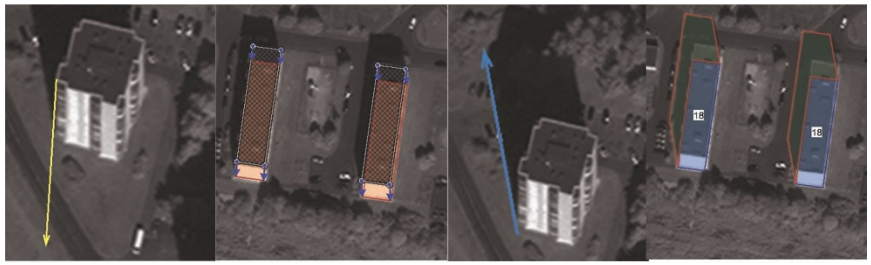
The method to reproject buildings into the true ground positions is designed to correct buildings and structures' positions on the rectified non-nadir space imagery. The *a priori* data — the direction of reprojection — is a vector in two-dimensional coordinates which is



(a)

(b)

Figure 23.5. Results of automatic digitizing of vegetation (a) and hydrography (b).



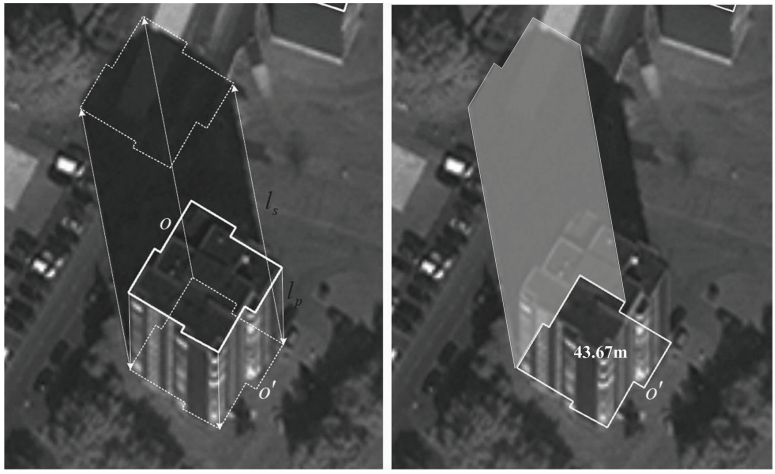
(a) (b) (c) (d)

Figure 23.6. Direction of objects' reprojection (a), result of reprojection (b), direction of objects' shading (c), and result of shadows detection, indicating calculated heights (d).

assumed to be constant for the entire space image (Figure 23.6(a)). Reprojection of an object is performed along the specified vector at a distance calculated as the best correspondence of its vector's shape to the base (Figure 23.6(b)). Shadow detection is intended to calculate the heights of the corresponding objects using the azimuth and elevation of the Sun during the image creation period. The azimuth of the Sun determines the direction of shading on the image (Figure 23.6(c)), the elevation angle is used to calculate the height of the object along the length of its shadow (Figure 23.6(d)).

The method is based on the ability to determine the height of building on a space image, both by its shadow and by the visible part of the walls. On account of correlation between the length l_p of walls' projection with its shadow length l_s (Figure 23.7(a)) this method allows simultaneously and automatically to transfer vector representation of the building o to the true ground position o' and calculate its height (Figure 23.7(b)). To detect the shadow, we used the first derivative of the Gaussian function [44] oriented along the reprojection and shading directions.

For the experimental estimation of the precision and time consumption of the developed method, an image's pair representing the single terrain site obtained at different survey angles was used. In both images, 924 buildings were digitized. The reference digital map is based on OSM [28] data.



(a) (b)

Figure 23.7. Scheme (a) and result (b) of transferring the building's vector representation to the true ground position and calculating its height.

Within the experiment framework reprojection and calculation of buildings' heights were carried out in the following ways:

- manually by the operator (only on the first image, as buildings bases on the second one are situated in shades and cannot be visually found),
- in automatic mode by the developed method on both images,
- in automated mode by the developed method with manual error correction on both images.

The evaluation criteria are as follows:

- mean points distance of the reprojected vector from the reference one (RMSE),
- difference between calculated and reference buildings' height (HE),
- total process time and average time per object.

Results of the experiment are given in Table 23.2.

Results of the experiment demonstrate that the implementation of the developed method in automatic mode allows reducing time costs

Table 23.2. Mean values of speed and precision of reprojection and height calculation.

Method	Stage	Time, min	Time per object, sec	RMSE, pixels	HE, pixels
Manual, Image1 1	Reprojecting	91	5,91	3,72	—
	Height calculation and correction	345	22,40	2,65	**1,03**
	Total	436	28,31		
Automatic, image1		84	5,46	2,13	1,67
Automated, image 1		105	6,82	2,05	1,40
Automatic, image 2		**80**	**5,20**	1,40	1,73
Automated, image 2		105	6,43	**1,34**	1,29

by 5 times and at least by 20% to increase the precision of reprojection. The developed method in automated mode allows reaching the precision of calculating the height equivalent to the manual method at 4-time superiority in time consumption.

23.3 Methods of Processing and Recognition for Raster Topographic Maps

Raster topographic maps are the least expensive data source to construct the DSM because, for the scale of 1:25 000 and larger, they are freely available. The complexity of the map-to-DSM transformation is in the necessity to digitize the contours, to assign them elevation values, and to set height markers' positions, which without the automation are executed with the interactive software tools and entail significant time costs (from 1 to 8 days per one map sheet, depending on the complexity of the relief).

DSM is constructed from the following elements of the topographic map:

(1) The contour (isoline) of relief is a closed line on the map with all located on the same elevation. The elevation difference between adjacent contours is called the isoline step and is determined by the scale of the map and relief (elevation range) of displayed

terrain. On the maps there are two types of contours: thickened, multiple to the five-fold step of the isoline (that is, every fifth contour is thickened), and the main ones. The contours on the topographic maps have the following types:

- the land relief contours (Figure 23.8(a)),
- land relief contours located in the regions of glaciers and firn snow (Figure 23.8(b)),
- coastlines of constant and variable hydrography (Figure 23.8(c)),
- contours of the bottom relief (Figure 23.8(d)).

(2) Elevation sign is a sequence of digits representing the elevation value for a point or contour in meters (for Russian maps). The following types of elevation signs are available:

- elevation mark is the elevation value at a certain point, as a rule, on top of the hill, or at a reference point (Figure 23.8(e)),
- a water edge mark is the elevation value within the river bed (Figure 23.8(f)),
- the contour (isoline) sign is an elevation value for contour and is placed in its discontinuity (Figures 23.8(g) and 23.8(h)).

(3) Permanent hydrography is "slow water" of constant height (elevation within the mirror of water is unchanged): lakes, reservoirs, seas, and oceans:

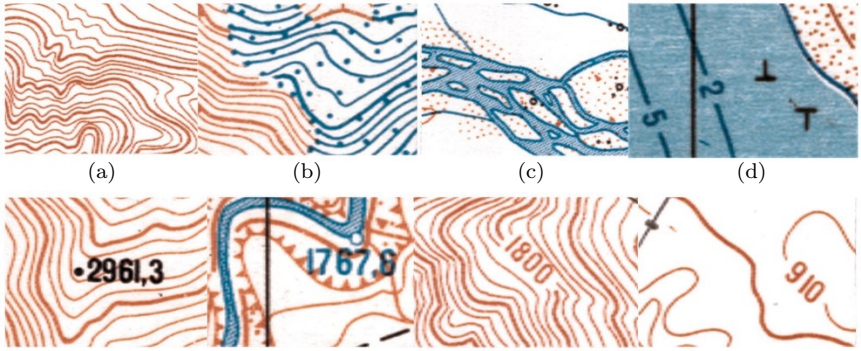


Figure 23.8. The main relief elements of topography maps.

The microrelief markers (gullies, sandy mounds, etc.) describe elevation variations within one isoline step and, to a small extent, supplement the relief formed with the elements described above.

Now there are a number of available software tools for the automated processing of raster topographic maps allow us to reduce time costs by 2–5 times compared to the interactive approach. The most functional of them at the moment — EasyTrace software package — simplifies the operator's work in terms of digitizing contours which makes topographic maps the most attractive source of data for DSM due to the best ratio of resulting data precision to the cost of digitizing [10].

However EasyTrace complex has a number of disadvantages that prevent further reduction of DSM construction process time (up to several hours):

(1) It is not possible to split automatically adjacent contours in case of high contour density.
(2) There are no instruments to recognize the elevation value signs and automatically assign elevations to the detected contours.
(3) There is no automated detection and elimination of microrelief overlapping the contours.
(4) Operator of the complex must have a sufficiently high qualification to ensure required precision of the resulting DSM.

The authors developed methods to process and recognize the raster topographic maps which provide an automatic construction of DSM and include:

- method to create a set of contours' vector representations (digitizing),
- method to detect, recognize, and classify the elevation signs,
- method to assign the elevation values to contours and to correct the possible recognition errors.

It should be noted that all methods for Russian maps are performed automatically without an operator, as for maps of other countries a preliminary selection of the color scheme, checking, and replenishment of the drawing standards' database for elevation values are required. During the verification and error correction phase operator

is able to perform additional control routines to improve the precision of the resulting DSM which increases the processing time of one map sheet by up to two hours.

Let us consider developed methods more precisely.

23.3.1 *Method to digitize contours*

EasyTrace performs automated digitizing of contours using color classification techniques for topographic map points leaving only brown and blue colors. This method is easy to implement, however, in case of a small distance between the contours (Figure 23.9(a)), the presence of a microrelief (Figure 23.9(b)), urban building elements or the road network (Figure 23.9(c)), it is required to make a long manual correction to separate the contours (Figure 23.9(d)) or remove objects which are not contours but have the same color (Figures 23.9(e) and 23.9(f)).

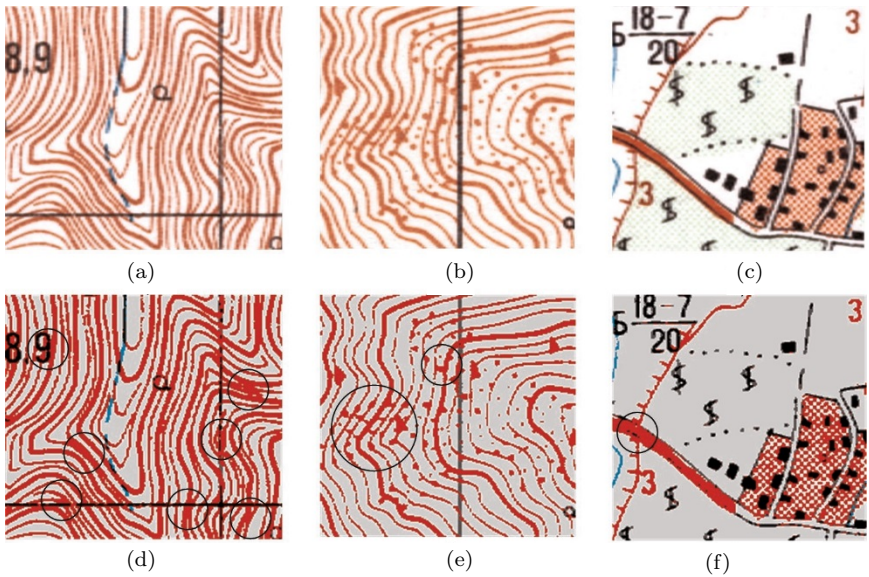


Figure 23.9. Defects of contours in color segmentation caused with their high density (a, d), presence of microrelief signs (b, d) and intersection with the road (c, f) (the areas of defects are marked by circles).

The authors developed a method to detect the center of the contour on a brighter (comparing with isolines) background which is based on smoothing the topographic map with a low-pass filter with a pronounced maximum in its center and subsequently constructing a map of local minima of the brightness function and its color segmentation.

A low-pass filter (Gauss filter [44]) provides partial transmission of its form to the gently sloping sections of the image brightness function which makes it possible to transform a uniform contour's brightness to the shape having the local minimum in central part, and also suppress local defects caused by errors in the analog-to-digital conversion during scanning of topographic maps.

A binary map of linear objects is constructed by searching for local brightness minima on the smoothed raster topographic map which is used for their vector encoding [16]. The transformation of linear objects into a vector representation provides their further analysis and assignment of elevation values and also facilitates the search and elimination of minor defects, like gaps and branches of 1–3 pixels in length.

The generated vector representations for linear objects are arranged into a number of layers by his segmentation [15] which replaces brightness and color of the current point with the values mean of the nearest color class from a classification set. This set for Russian maps is given a priori, for maps of other countries it is created when the color scheme is being customized. We distinguished layers corresponding to the brown and blue colors as containing the horizontals.

In both layers, we search for linear objects with the width and brightness corresponding to thickened contours. Thickened and main contours are handled independently that provides fewer errors by

- restoration of gaps (a thickened line cannot be mistakenly combined with the main one),
- assigning heights to the contours (the elevation of the thickened contour is always a multiple of the five-fold step of the contour, the main one is not).

23.3.2 *The method for elevation values recognition*

The method for automatic recognition of elevation values on raster topographic maps consists of the following stages:

(1) Detection and vector encoding of objects' boundaries related to black (elevation mark), brown (the contour sign), and blue (water edge mark) colors,
(2) Determining wherever each detected object belongs to the class of digits, recognizing the value, and extracting its geometric parameters,
(3) Composing the digits sequences, calculating the elevation values, and final semantic verifying.

Single-color objects' detection on a topographic map is performed by their segmentation in the HSI color space (result of segmentation of the map sample in Figure 23.10(a) is presented in Figure 23.10(b)). A segmented topographic map image is further transformed into a set of clusters representing spatially isolated homogeneous regions bounded by the points of any other color.

To determine the fact that the cluster belongs to the digits class we need a mathematical apparatus that allows determining the degree of object's similarity to a reference and is invariant to their mutual scale and rotation angle which appears according the following reasons:

- digits of elevation signs within one map may differ in size up to two times (for instance, so-called "commander's" heights),
- contour signs are located on the map at an arbitrary angle which denotes the direction of elevation gradient,
- there are a number of different shapes of elevation markers to be recognized (points of the elevation value geographic location.

The mathematical apparatus of contour analysis [14] is chosen to solve the problem of elevation signs recognition which provides determination of similarity degree of cluster's contour for each of the digit references ("0"–"9") and elevation markers regardless of their mutual scale and rotation angle. The use of contour analysis for a contour

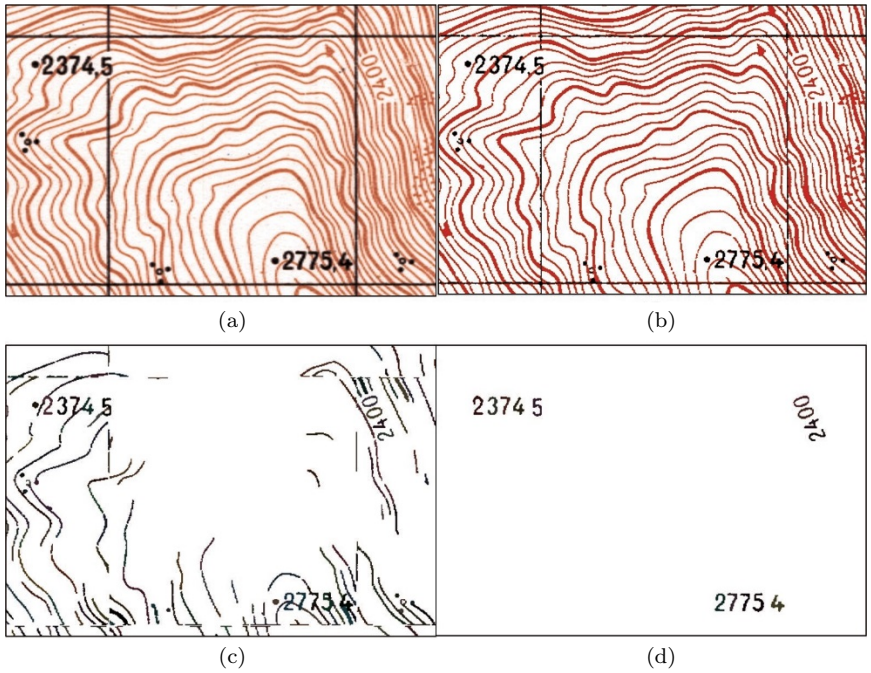


Figure 23.10. Color segmentation of the topographic map (b), digits recognition (c) and sequences composing (d).

requires its transformation into a complex-valued vector by encoding boundaries with adjacent clusters of other colors.

If the cluster has a low similarity degree (<65%) for all references it is to be not a digit and is not considered further at all, otherwise it is used to compose sequences (clusters remained after matching are shown in Figure 23.10(c)) representing the elevation values. A large number of contour fragments recognized as digits in Figure 23.10(c) are explained by their similarity with digits "1" and "7".

Sequences of adjacent digits are considered as the elevation value if the following requirements are met:

(1) All digits must have the same color, difference's module in angles from the y-axis should not exceed 5°, and the values of digits' heights calculated at this angle should differ by at most 15%.
(2) Centers of rectangles containing digits should be located on the same straight line (with allowable deviation radius not more than

10% of the digit height) at equal distances between their boundaries (the exception is the distance between the integer and the fractional part of the elevation value).

Additionally the semantic verification is performed:

(1) Elevation value should be within [0.8848] meters.
(2) Digits of elevation or water edge mark should have a negligible (not more than 15°) deviation from the vertical.
(3) Value of contour sign must be multiple of the isoline step.

Digit sequences that satisfy the above-mentioned requirements are presented in Figure 23.10(d).

Recognized elevation values should be assigned to specific objects that determine their position on the topographic map and on the presented terrain. Thus for elevations and water edge marks a search for insignificantly remote elevation markers is made. For each found marker a point object with a corresponding elevation value in the center is created.

The elevation values given by contour signs are assigned to the corresponding linear objects in which discontinuity they were found. In case of several adjacent lines, the choice is made depending on the sign correspondence to the thickened or main contour.

As a result of applying the described methods, a vector DSM is created, containing contours in the form of linear objects and elevation marks in the form of point objects, and elevation values are assigned only to signed objects recognized on the original raster map.

23.3.3 *The method to determine the contours' elevations and correct the recognition errors*

The method to determine the contours' elevations and correct the recognition errors is based on the consciousness of an oriented graph

for objects with known elevations by triangulation using the Delaunay criterion [36]. Elevation values correspond to vertices of the graph, the weights of edges represent the number of contours between the elevation marks and orientation of the edge indicates the direction of elevation decrease.

Recognition error control is based on the fact that the weights of the edges must correspond to the number of contours that are crossed by the segment connecting elevation marks on the topographic map. If these values do not coincide the estimates of recognition error for the vertices incident to this edge is increased by one. Thus vertexes or vertex groups which have higher evaluations than adjacent ones are considered to be erroneous and do not participate in the assignment of elevations to the contours. Also for error control a low-detail entire-world DSM from SRTM with $3''$ step [33] is used.

The elevation assignment to the contours is performed in a manner similar to error control. The elevation values calculated by a sequential searching for the intersections of contours by an edge enclosed between each pair of vertices.

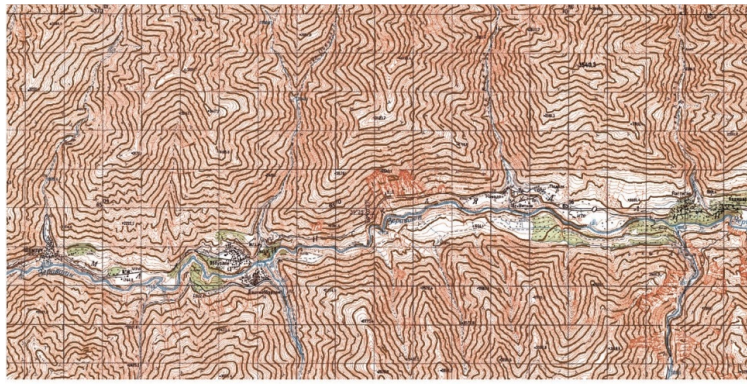
In case an operator needs to verify the precision of the relief vector map he has the ability to export it to GIS Map, MapInfo, ArcGIS and then to search and correct errors of contours detection and elevations calculation using vector editing toolset.

Resulting regular DSM in the form of elevation matrix is formed on the basis of vector map with standard triangulation algorithms [36] implemented in most modern GIS and allowing to set the resolution and scale of the output model (Figure 23.11).

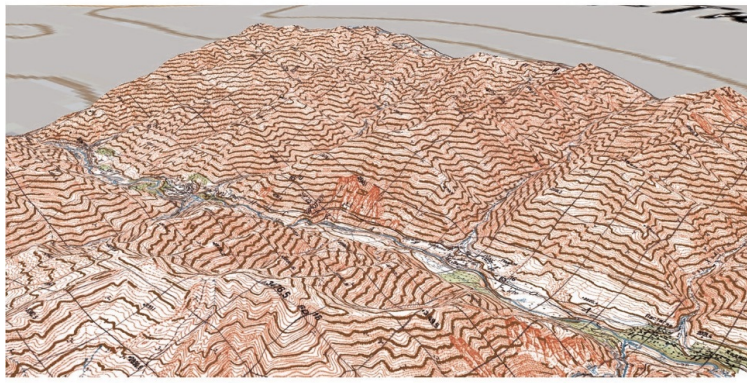
Developed methods ensure automatic construction of DSM from a raster topographic map, and for the processing time of a single map sheet scanned at 400 dpi, regardless of the scale, does not exceed 10 minutes (for workstation with Intel processor Core i5 2400, 8GB RAM) and varies according the number of relief elements and its density. The experimental results for precision and time to create a DSM comparing to the analogs are given in Table 23.3 (the map created in GIS Panorama).

Table 23.3. Time and precision values for DSM creation with the developed method comparing the analogues.

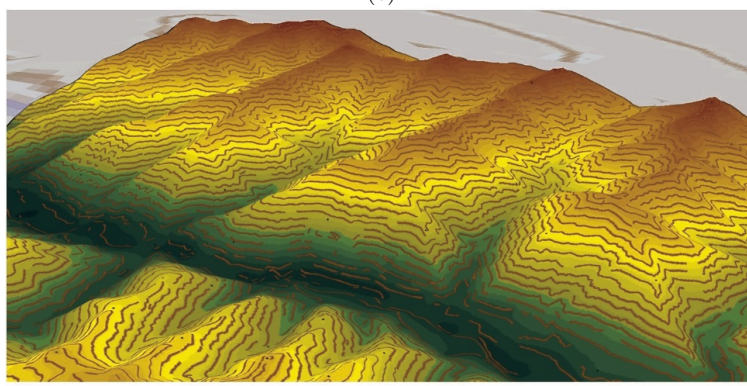
Map number	Isoline step, m	Elevation range, m	Number of contours	Number of elevation marks and contour signs	Software	Mean error of contours localization, points	Mean error of height determination, m	Creation time of DSM, hour
1	20	1830–3845	342	68	GIS Panorama	0	0	82.1
					Easy Trace	1.52	4.92	18.8
					Developed	2.86	29.19	**0.14**
2	10	277–2944	270	51	GIS Panorama	0	0	61.3
					Easy Trace	1.37	2.73	12.2
					Developed	2.22	17.41	**0.13**



(a)



(b)



(c)

Figure 23.11. Source raster map with thickened contours overlaying (a), three-dimensional view of resulting DSM with thickened contours overlaying (b) and elevation model in pseudocolor visualization (c).

References

[1] Ballard, D.H. (1981) Generalizing the Hough transform to detect arbitrary shapes. *Pattern Recognition*, 13(2), 111–122.

[2] Blundell, J.S. and Opitz, D.W. (2006) Object recognition and feature extraction from imagery: The Feature Analyst® approach. *International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences*, 36(4), 42–47.

[3] Blundell, S., Opitz, D., Morris, M., and Rao, R. (2008) Feature analyst v5.0. *Proceedings of the ASPRS Annual Conference*, pp. 28–37.

[4] Burt, P. and Adelson, E. (1983) The Laplacian pyramid as a compact image code. *IEEE Transactions on Communications*, 31(4), 532–540.

[5] Caffe. (2018). Deep learning framework by BAIR. http://caffe.berkeleyvision.org/. Accessed 12.10.2018.

[6] Casanova, D., Florindo, J.B., Falvo, M., and Bruno, O.M. (2016) Texture analysis using fractal descriptors estimated by the mutual interference of color channels. *Information Sciences*, 346, 58–72.

[7] Complex for automated recognition and digitizing of remote sensing data (in Russian). https://gisinfo.ru/products/automap.htm. Accessed 12.10.2018.

[8] Dremov, E.N. and Miroshnichenko, S.Y. (2017). The method to detect a subset of thickened isolines on raster topographic maps. *Telecommunications*, 6, 37–42 (in Russian).

[9] Dstl Satellite Imagery Feature Detection. https://www.kaggle.com/c/dstl-satellite-imagery-feature-detection. Accessed 12.10.2018.

[10] Easy Trace. What is it, and what for? http://www.easytrace.com/program/about_ru. Accessed 12.10.2018.

[11] Enhanced Vegetation Index. https://en.wikipedia.org/wiki/Enhanced_vegetation_index. Accessed 12.10.2018.

[12] ENVI EX User's Guide. http://www.harrisgeospatial.com/portals/0/pdfs/enviex/ENVI_EX_User_Guide.pdf. Accessed 12.10.2018.

[13] Flach, P. (2012) *Machine Learning: The Art and Science of Algorithms That Make Sense of Data* (Cambridge University Press, Cambridge, UK).

[14] Furman, Y.A., Krevetskii, A.V., and Peredreev, A.K. (2002) *An Introduction to Contour Analysis: Applications to Image and Signal Processing* (Fizmatlit, Moscow) (in Russian).

[15] Gonzalez, R. and Woods, R. (2002) *Digital Image Processing* (Pearson Education, Upper Saddle River, New Jersey, USA).

[16] Gorbachev, S.V., Emelyanov, S.G., Zhdanov, D.S., Miroshnichenko, S.Y., Syryamkin, V.I., Titov, D.V., and Shashev, D.V. (2018) *Digital Processing of Aerospace Images* (Red Square Scientific, Ltd., London, UK).

[17] Gurney, K. (2014) *An Introduction to Neural Networks* (CRC Press, Boca Raton, Florida, USA).

[18] How we participated in SpaceNet three Road Detector challenge. And how we got into top 10. https://spark-in.me/post/spacenet-three-challenge/. Accessed 12.10.2018.

[19] Iglovikov, V. British Satellite Images 2: How It Was Really (in Russian). https://habrahabr.ru/company/ods/blog/330118/. Accessed 12.10.2018.

[20] Iglovikov, V., Mushinskiy, S., and Osin, V. (2017) Satellite imagery feature detection using deep convolutional neural network: A Kaggle competition. arXiv: 1706.06169 [cs.CV].

[21] Jebur, M.N., Mohd Shafri, H.Z., Pradhan, B., and Tehrany, M.S. (2014) Per-pixel and object-oriented classification methods for mapping urban land cover extraction using SPOT 5 imagery. *Geocarto International*, 29(7), 792–806.

[22] Kaggle: How Our Networks Counted the Sea Lions in the Aleutian Islands (in Russian). https://habrahabr.ru/company/ods/blog/337548/. Accessed 12.10.2018.

[23] Kuzin, A. The Second Honorable Place. Notes of the Satellite Imagery Feature Detection Challenge Participant (in Russian). https://habrahabr.ru/company/avito/blog/325632/. Accessed 12.10.2018.

[24] Lee, K. Dstl Satellite Imagery Competition, 1st Place Winner's Interview. http://blog.kaggle.com/2017/04/26/dstl-satellite-imagery-competition-1st-place-winners-interview-kyle-lee/. Accessed 12.10.2018.

[25] Manual for Satellite Data Analysis eCognition Developer. http://open_jicareport.jica.go.jp/pdf/12150314_03.pdf. Accessed 12.10.2018.

[26] MapInfo Pro — Desktop GIS. https://www.pitneybowes.com/us/location-intelligence/geographic-information-systems/mapinfo-pro.html. Accessed 12.10.2018.

[27] Modeling, Documentation, and Visualization Software. https://www.bentley.com/en/products/brands/microstation. Accessed 12.10.2018.

[28] NextGIS Data. Geodata in ESRI Shape, ESRI Geodatabase, GeoJSON, Mapinfo Tab. https://data.nextgis.com/en/. Accessed 12.10.2018.

[29] NOAA Fisheries Steller Sea Lion Population Count. https://www.kaggle.com/c/noaa-fisheries-steller-sea-lion-population-count. Accessed 12.10.2018.

[30] Normalized Difference Water Index. https://en.wikipedia.org/wiki/Normalized_difference_water_index. Accessed 12.10.2018.

[31] Obusek, F. Delineating Rooftops in ERDAS Imagine. https://www.youtube.com/watch?v=HBcPB6Agr5c. Accessed 12.10.2018.

[32] Professional GIS “Panorama” (in Russian). https://gisinfo.ru/products/map12_prof.htm. Accessed 12.10.2018.
[33] Reuter, H.I., Nelson, A., and Jarvis, A. (2007) An evaluation of void-filling interpolation methods for SRTM data. *International Journal of Geographical Information Science*, 21(9), 983–1008.
[34] Safe Passage: Detecting and Classifying Vehicles in Aerial Imagery. https://www.datasciencechallenge.org/challenges/1/safe-passage. Accessed 12.10.2018.
[35] Schowengerdt, R.A. (2007) *Remote Sensing: Models and Methods for Image Processing* (Elsevier, Amsterdam, Netherlands).
[36] Sloan, S.W. (1987) A fast algorithm for constructing Delaunay triangulations in the plane. *Advances in Engineering Software*, 9(1), 34–55.
[37] Soil-Adjusted Vegetation Index. https://en.wikipedia.org/wiki/Soil-Adjusted_Vegetation_Index. Accessed 12.10.2018.
[38] Tensorflow. (2018) An open source machine learning framework for everyone. https://www.tensorflow.org/. Accessed 12.10.2018.
[39] The open competition for a best solution in a field of intellectual technologies creation for recognition of aerospace imagery in visible spectrum (in Russian). http://fpi.gov.ru/activities/konkurs/spacemap. Accessed 12.10.2018.
[40] Theano. http://deeplearning.net/software/theano/. Accessed 12.10.2018.
[41] Torch. (2018) A scientific computing framework for LuaJIT. http://torch.ch. Accessed 12.10.2018.
[42] Use of the Canopy Chlorophyl Content Index (CCCI) for Remote Estimation of Wheat Nitrogen Content in Rainfed Environments. https://www.researchgate.net/publication/259360047_Use_of_the_Canopy_Chlorophyl_Content_Index_CCCI_for_Remote_Estimation_of_Wheat_Nitrogen_Content_in_Rainfed_Environments. Accessed 12.10.2018.
[43] Varma, M. and Zisserman, A. (2003) Texture classification: Are filter banks necessary? *Computer Vision and Pattern Recognition*, 2, II–691.
[44] Vliet, L.J. and Verbeek, P.W. (1994) Edge localization by MoG filters: Multiple-of-Gaussians. *Pattern Recognition Letters*, 15(5), 485–496.
[45] Zhangm, Y., Zhang, H., and Lin, H. (2014) Improving the impervious surface estimation with combined use of optical and SAR remote sensing images. *Remote Sensing of Environment*, 141, 155–167.

© 2025 World Scientific Publishing Company
https://doi.org/10.1142/9789811267215_0024

Chapter 24

Mobile ID Recognition: Coarse-to-Fine Approach

Vladimir L. Arlazarov[*,‡,§], **Vladimir V. Arlazarov**[*,‡,‖], **Konstantin B. Bulatov**[*,‡,¶], **Timofey S. Chernov**[‡,**], **Dmitry P. Nikolaev**[†,‡,††], **Alexander V. Sheshkus**[*,‡,‡‡], **Natalya S. Skoryukina**[*,‡,§§], **Oleg A. Slavin**[*,‡,‖‖], **and Sergey A. Usilin**[*,‡,¶¶]

[*]*Federal Research Center Computer Science and Control of the Russian Academy of Sciences, Moscow, the Russian Federation*
[†]*Institute for Information Transmission Problems of the Russian Academy of Sciences (Kharkevich Institute), Moscow, the Russian Federation*
[‡]*Smart Engines Service LLC, Moscow, the Russian Federation*
[§]*vladimir.arlazarov@smartengines.com*
[‖]*vva@smartengines.com*
[¶]*kbulatov@smartengines.com*
[**]*chernov.tim@smartengines.com*
[††]*dimonstr@iitp.ru*
[‡‡]*asheshkus@smartengines.com*
[§§]*skleppy.inc@smartengines.com*
[‖‖]*oslavin@smartengines.com*
[¶¶]*usilin@smartengines.com*

Automatic optical recognition of documents is a traditional function of modern document processing systems. In this field, "recognition" means complex process which includes image processing, segmentation, classification, and linguistic analysis. Although the idea of using mobile devices

for recognition of paper documents is not new, direct usage of existing software solutions for scanned images recognition yields low recognition precision on images obtained using a mobile device. This is due, first of all, to perspective distortions and lower effective resolution in the latter case. Indeed, one of the most widespread approaches to document images recognition is identification and recognition of arbitrary text on images with subsequent text result analysis. Following this, document classification and field attribution require recognition of document template text which is usually much smaller in size than the filling text and therefore almost unreadable on mobile images. Super-resolution algorithms allow us to obtain a single better quality image, but this approach is too resource-consuming in a sense that it implies improvement of the whole document image, whereas for a large number of document types, it would be enough to focus just on several field regions. In this chapter, we present an original approach and algorithms for recognition of video frame sequence containing a document image, which is applicable for mobile implementation. It is based on the coarse-to-fine methodology, where template matching and field localization are performed on the image with lowered resolution, followed by lazy processing of parts of the images only corresponding to the fields which are not recognized yet. Video stream is utilized as a source of noise reduction both in coordinates of the fields and OCR classifier outputs. The algorithm based on the proposed approach is suitable for running on the device itself and can operate even when none of the video frames contain a document image of sufficient quality by themselves.

Keywords: OCR, optical document recognition, coarse-to-fine recognition, AI mobile application, real-time video processing, document identification and matching, anytime algorithms, data augmentation, neural networks optimization.

24.1 Introduction

Smartphones and other mobile devices have become the *de facto* way of receiving various government and commercial services, including but not limited to e-government, fintech, banking, and sharing economy [59]. Most of them require the input of personal data [44]. Unfortunately, manual entry of document data via mobile phone or tablet can sometimes be inconvenient, time consuming, and error-prone. Therefore, many organizations involved in these areas decide to utilize identity document analysis systems in order to improve data input processes. The goals of such systems are to perform

document data recognition and extraction, to prevent identity fraud by detecting document forgery or by checking whether the document is genuine and real, and others. Modern systems of such kinds are often expected to operate in unconstrained environments.

The idea of using mobile devices for digitization and recognition of paper documents is not new (e.g. [49,64]). For a number of years, low quality of small-scale digital cameras and insufficient computational capabilities restrained the achievement of serious practical results. However, technology improvements in the last few years provided the opportunity to create document recognition systems which will work on mobile devices and will have industrial precision and performance [4,25].

The three major goals of the mobile document recognition system are as follows:

(1) *Accuracy*: This should go without saying that even though the documents have to be recognized on a mobile device and in photos or video frames produced with small-scale cameras, the recognition accuracy must be high in order for this system to be a viable way of data entry.
(2) *Speed*: One of the main disadvantages of manual document data entry on mobile devices is the time and concentration required to perform such tasks. The integration of document recognition in business processes should be seamless, so one of the most important requirements for mobile document analysis systems is its execution speed. This becomes more complicated with the constraint of autonomy, considering the limited computational capabilities of mobile devices.
(3) *Scalability*: It is natural to assume that the highest recognition accuracy can be achieved if the system is specifically tuned to a set of pre-defined document form templates. However, it is important for the system to be designed in such a way that the addition of new types of documents or new templates can be performed as easily as possible. Furthermore, obtaining real samples of identity documents for the training dataset is complicated and sometimes impossible, thus the number of samples required for the addition of new document template should be minimal.

The two main challenges of mobile document recognition systems are limited computational resources available on mobile devices and

the specific set of image distortions, characteristic to small-scale digital cameras, which may significantly diminish the recognition quality. The most common image distortions associated with small-scale mobile device cameras which influence the recognition process are as follows [10]:

- Digital noise: Signal quantization errors, thermal noise, and charge-transfer noise. Additional sources of distortions are matrix pollution, sensor element defects, and image compression algorithms, especially notable for video stream frames. Digital noise often occurs in low illumination conditions.
- Perspective (projective) distortions of the document image: An assumption of the document elements being parallel and perpendicular to each other (e.g. lined paper template and text lines) is often used in optical text recognition algorithms (see [95]). However, it is obvious that using a digital video camera to capture a document, the angles on the resulting images and objects' proportions change depending on the aspect angle. Due to this fact, classical algorithms become inapplicable without preliminary projective normalization.
- Non-linear distortions of the document image due to bending and folding of the document pages.
- Defocus and motion blur due to video camera displacement during exposition.
- Highlights and shadows due to uncontrolled lighting conditions or specific document features, such as holographic artifacts [37].

Due to such distortions, direct usage of existing software solutions for scanned images recognition yields low recognition precision on images obtained using a mobile device. In addition, modern mobile device recognition use cases usually involve unknown and uncontrolled environment and a person who handles the recognition system often is not familiar with how such system operates. Uncontrolled or unconstrained environment means that scene geometry, lighting conditions, and relative movement model of the object and the capturing device are unknown. It also implies that there is no guarantee for the input data to be of high quality and indicates the possible risk of information loss. If the video of the document is captured using a mobile device, then, depending on which particular frame has been

captured, the document data information might be partially or completely lost because of the document not being fully present inside the frame, camera not being fully focused, etc. [35].

The possible solution which addresses the aforementioned problems is to recognize the whole video stream instead of only one single frame. One possible approach is to use a sequence of video frames to obtain a single better quality image (see [69, 87]). This approach can be too resource-consuming and global in the sense that it requires restoration of improved image for the whole document, whereas for a large number of document types, we can only focus on localized fields regions. A number of commercial recognition systems use client–server schemes ("cloud recognition"), in which the input image is transmitted from a mobile device to a remote server which performs main processing and recognition (see [72,139]). These schemes allow using more resource-consuming image processing algorithms on server side but present disadvantages such as requirement of constant network connection during operation and possible security issues when dealing with transmission of sensitive documents data.

Problems related to mobile document image capturing aside, it is also important to take into account special properties of identity documents. Identity documents hold sensitive personal data which also makes the cost of recognition errors quite high compared to common text recognition tasks. In order to provide additional level of forgery protection, identity documents often have complex graphical background possibly containing guilloche, watermarks, retroreflective coating which is prone to highlights, and holographic security elements which change their appearance depending on relative positions of camera and document [37]. Moreover, text is printed with special fonts, sometimes using indent or embossed printing techniques. Another feature of identity documents is their field layout, which is often strict in terms of absolute and relative position and size of document elements. However, there could be complex exceptions, such as optional text fields which may or may not be present on particular document, floating textual and graphical elements and so on.

In this chapter, an overview is presented for an original approach and set of methods for the recognition of video stream containing images of identity documents, which is suitable for realization on mobile devices. The approach is based on a "coarse-to-fine" methodology: at each step, the partial information is extracted in

low resolution, only to minimize the amount of information to be processed on later stages. Algorithms presented in the scope of this overview are suitable for execution on mobile devices without exploitation of external computing power and are capable of analyzing video frame sequences in which no single frame contains a high-quality image of a full document.

24.2 Overview

Automatic recognition of digitized documents is a traditional function of modern document processing systems. By "recognition" a complicated iterative process is generally meant, which includes different algorithms of segmentation, classification, and linguistic analysis. The problem statement for printed document recognition on a mobile device is a classical one and entails document edges finding (or document location in a broader sense), matching using markers (such as lines and static text), zones, and field location and recognition of symbols of each field. Description of the document may be significantly complicated and contain document geometry (size and proportions), fiducial elements (their sizes, proportions, text contents, etc.), and characteristics of symbols and fields (font, size range, dictionaries of common words, finite state language models, etc.). In this chapter, a system is presented which relies on an structured sequence of methods which solve separate image processing tasks, representing a "coarse-to-fine" document understanding approach [3]. The main algorithm scheme is presented in Figure 24.1.

Frame quality evaluation: Input images of optical recognition systems can be subjected to a large number of different distortions, especially in uncontrolled or natural capturing conditions [10]: defocus, blur, digital noise, compression artifacts, highlights, projective distortions, damage to the document object, and so on. Such distortions result in information loss and therefore may have a negative effect on each of the following image processing tasks. In order to address these distortions and reduce their impact on the recognition system, the image quality assessment methods are used on every processing step along with the methods for image quality reconstruction, recognition confidence control, and feedback [35, 36, 110].

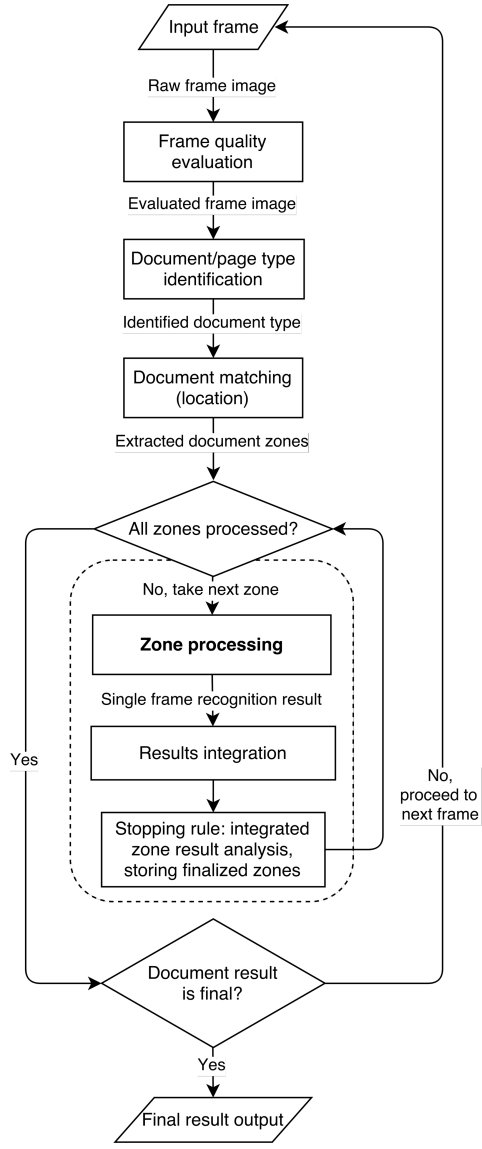


Figure 24.1. Main frame processing algorithm scheme. More detailed scheme of zones processing (dashed region) is presented in Figure 24.2.

Document identification and matching: Detection of a document on a video frame, document type identification, and page location on a frame is possible without text recognition. In the

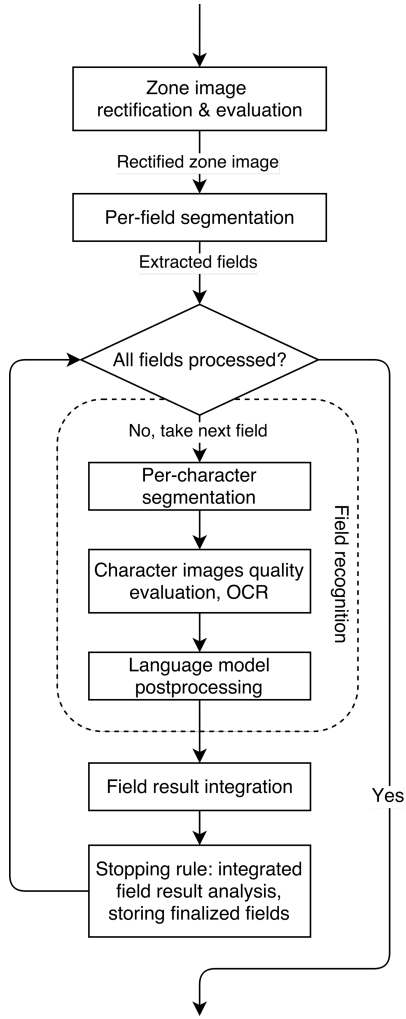


Figure 24.2. Document zone processing scheme.

implemented system, depending on the class of documents, a series of methods are used: generalized Viola–Jones method as a decision tree of strong classifiers [92, 133] for robust document page detection and location with moderate perspective distortions and for fiducial elements location (e.g. logo images or personal photos). A more universal method uses document identification and matching using feature points detection [12, 81, 86, 118] with descriptors

indexing [89, 90, 131] and their geometric positioning analysis using robust RANSAC implementation [56,108,111] and fast Hough transform [98], and rectangular documents image candidates search using fast Hough transform and combinatorial analysis [120, 128].

A more comprehensive description of the methods for document identification and location is given in Section 24.4.

Zone processing and per-field segmentation: Let us define a "document zone" (see Figure 24.3) as a region of a document page containing a localized set of fields which can be processed by a single application of per-field segmentation algorithm (an algorithm of separate fields localization). In order to minimize the computation needed for perspective correction of the frame, the image of each zone, which is not obstructed by highlights or other signal distortions, is rectified separately. Field localization is implemented using morphological analysis of document zones with known structure. Irregularities and roughness of document matching algorithms are balanced by the robustness of per-field segmentation algorithm due to preliminary text line angle detection using fast Hough transform.

The majority of automatic document entry systems rely on text recognition with standard typesetting, but in practice, italic and handwritten text fields occur on identity documents in a significant scale, which leads to complications of standard document field processing scheme. This problem may be simplified by preliminary identification of these situations and normalization of text fields to a canonical form. A solution implemented in the described algorithm is a shear transform application to an input field image, given the slant angle found using a text slant rectification algorithm based on fast Hough transform [82]. This method also adds an additional robustness to per-field segmentation algorithm.

More detailed description of zones per-field segmentation methods is presented in Section 24.5.

Per-character field segmentation and OCR: Once the individual text field lines were separated by the per-field segmentation, the per-character segmentation algorithm is required to obtain the positions (bounding boxes) of individual characters [31]. This problem has been studied for decades but still remains open [109]. Segmentation process is complicated by the variety of fonts used in identity

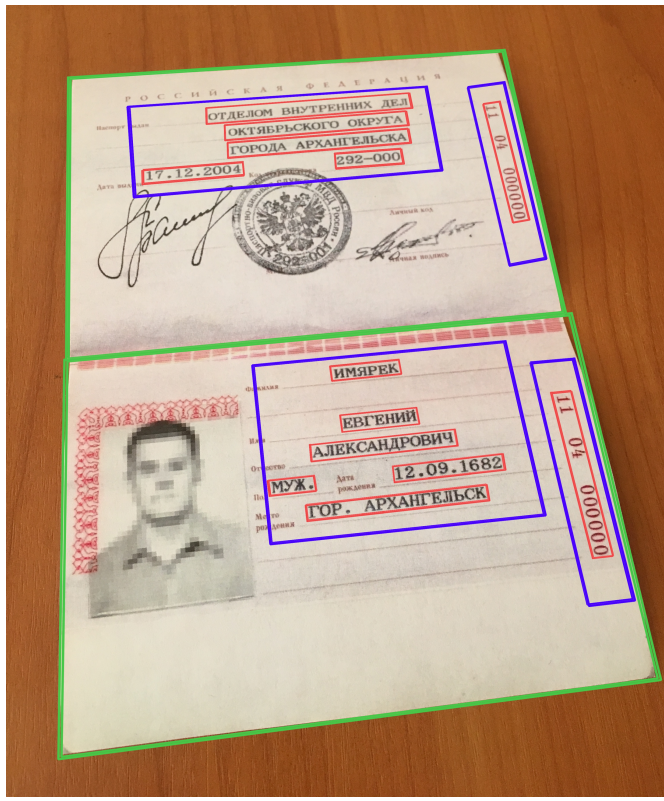


Figure 24.3. An example frame containing Internal Russian passport. Document zones (input for per-field segmentation) are highlighted in blue, and document fields are highlighted in red.

documents and also by the aforementioned capturing image distortions which may cause character breaking or gluing as well as other transformations negatively affecting the results.

Depending on document and field types, two main per-character segmentation methods are used which are described in detail in [34]. The first method develops classical approaches to text segmentation and comprises stages such as projection analysis, preliminary cutting, and dynamic programming with respect to character recognition pseudo-probability scores. The second segmentation method extensively utilizes machine learning approaches such as convolutional neural networks (CNN) and recurrent long short-term memory (LSTM) neural networks which reduce the number of heuristics

tied to specific document field types and also increases robustness to various distortions.

Given the large variation of distortions suffered by images of text characters even with optimal train dataset augmentation [65], the results obtained using a high-precision convolutional neural network for character recognition may be insufficient (e.g. on poorly focused characters or in the presence of highlights). Due to common neural network overconfidence problem even in those cases, the recognition results frequently have shifted class estimations distribution, thus the post-processing methods based on a field language model sometimes fail to correct these errors. It is possible to use preliminary character image quality evaluation such as image derivative analysis for contrast, defocus, and motion blur estimation with subsequent reweighting the relative impacts between primary hypothesis (neural network) and language model [28].

More detailed account of per-character field segmentation and character recognition is given in Section 24.6.

Integration and stopping rules: The system described in this chapter takes multiple video frames as inputs and thus operates in video stream in real time. While object recognition in video stream presents some advantages and helps solve problems related to camera-based image analysis, it creates new problems which have to be addressed, one of which is the problem of dealing with multiple recognition results of the same object obtained from different frames. This problem can be formulated either as the problem of optimal selection of the best result among the candidates obtained from different images or, more generally, as the problem of constructing an optimal integration function which will yield an accumulated result with the highest expected accuracy.

Additionally, a significant advantage of on-device video stream processing for document recognition is the possibility of organizing feedback between the low-level recognition procedures and the recognition system, and between the recognition system and the operator (user). The feedback could be roughly classified into three categories:

(1) *Informational feedback*: This includes visualization of the document geometry analysis (positions of found document pages, zones, and fields), visualization of fields recognition status and reliability estimation (predictor values), etc.

(2) *Advisory feedback*: This includes any information deduced by the system while processing the video stream which could be used to signal the operator or high-level process in order to improve the capturing process (refocus signals, camera flash control, a signal to change the camera angle to shift the highlights or hologram reflection position, etc.)
(3) *Recognition process control feedback*: This includes integrated field result analysis in order to determine when to stop the recognition process.

The crucial question for a real-time video stream recognition system is when to stop the recognition process (i.e. how to organize the recognition process control feedback). The simplest solution would be either to give the control to operator (and let the operator decide when to stop the recognition) or to break the recognition process after a fixed period of time. The first approach either leads to the operator stopping the recognition too early (because the operator is motivated to acquire the recognition results as soon as possible) or to stalling the recognition process due to the fields which cannot be recognized by the system for a reason not directly observed by the operator. The second approach ("time-out") leads to excessive time needed to recognize a document in good capturing conditions (when less time is actually needed to perform correct recognition) or to the opposite extreme — stopping the recognition process before the optimal point. This problem, along with the problem of multiple results integration, can be considered as the two most novel problems which occur in video stream recognition systems as opposed to the classical single-image analysis systems.

More detailed consideration of the integration and stopping problems is presented in Section 24.7.

24.3 Datasets

In order to perform research and development of systems for automated identity data entry, it is essential for researches to have access to relevant datasets. However, identity documents, as have been stated before, are special in a sense that they contain sensitive personal information. This creates complications in several aspects.

First, storing any personal data presents a security risk in case it is leaked, resulting in identity fraud and significant financial damages both for corresponding ID holders and the party responsible for leaking the data. Second, people understand that risk and are not as easily convinced to share their personal data with someone they do not trust. Third, identity documents are uniquely bound to their owners which makes them very rare, increasing the costs of data collection even further. Fourth, there are very few publicly available samples of each identity document and most of them are protected by copyright laws which makes them unusable in research. Finally, in many countries, it is illegal not only to distribute but even to collect and store personal data without a special permission [50].

For the reasons explained above, publicly available datasets containing identity document image are virtually non-existent. Research teams are forced to create and maintain their own datasets using their own resources. This can be easily observed by looking through the recent papers devoted to identity document analysis, for instance, [11, 45, 116]. For some teams, having access to such sensitive data is not a problem, especially if they are industry-oriented, already have a product related to identity document analysis, have government support, and possess the required expertise in data security. Commercial companies working in this field might receive large datasets from their customers, but they will never share it because of aforementioned legal, security, and also market advantage reasons. However, for most researchers, these options are not easily obtainable.

The absence of publicly available identity document datasets is a serious problem which directly affects the state of research in this field. It raises the entry barrier, putting off those who do not have the resources to create their own datasets and slowing down those who have not collected the data yet. Furthermore, it becomes impossible to evaluate and compare various identity document analysis methods to each other since they have been tested on completely different and locked down data [120, 133]. In addition, there is an ethical concern regarding verifiable and reproducible research, especially during peer review process. Even great papers presenting state-of-the-art results can be rejected by the scientific community for being impossible to verify without the testing data.

Some classes of publicly available datasets may be useful for researching common subtasks of identity document analysis and

Table 24.1. Related tasks and public datasets.

Task	Datasets
Document detection and localization	[29]
Text segmentation	[136]
Document image binarization	[106]
Optical character recognition	[79, 146]
Image super-resolution	[32]
Document forensics	[115]
Document image classification	[62]
Document layout analysis	[2, 42]
Document image quality assessment	[78, 96]
General image understanding	[46, 75]

recognition. Examples of such tasks and relevant datasets are listed in Table 24.1.

For the scope of problems related to a more general identity document analysis pipeline, however, these datasets prove insufficient, especially considering mobile video stream recognition systems. In this section, we present a Mobile Identity Document Video dataset (MIDV-500) [6], which in contrast to other relevant publicly available datasets can be used to develop, demonstrate, and benchmark a coherent processing pipeline of identity document analysis and recognition in its modern applications and use cases. All source document images used in MIDV-500 are either in public domain or distributed under public copyright licenses which allows us to use these images with attribution to the original publisher. The dataset is available for download at ftp://smartengines.com/midv-500/.

The MIDV-500 dataset contains video clips of 50 different identity document types, including 17 types of ID cards, 14 types of passports, 13 types of driving licenses, and 6 other identity documents of various countries. Each document image was printed on a photo paper using HP Color LaserJet CP2025 printer and laminated using glossy film. Then, for each document printout, we have recorded video clips in 5 different conditions using 2 mobile devices (10 videos per document in total). Conditions are described in Table 24.2.

The first letter of each identifier specifies the condition and the second letter specifies the mobile device on which the video clip was recorded. The two mobile devices used for recording were *A*pple

Table 24.2. Dataset structure.

Identifier	Description
TS, TA	"*T*able" — simplest case, the document lays on the table with homogeneous background
KS, KA	"*K*eyboard" — the document lays on various keyboards, making it harder to utilize straightforward edge detection techniques
HS, HA	"*H*and" — the document is held in hand
PS, PA	"*P*artial" — on some frames, the document is partially or completely hidden off-screen
CS, CA	"*C*lutter" — scene and background are intentionally stuffed with many unrelated objects



Figure 24.4. Example frames from MIDV-500 dataset. Five different conditions are presented (left to right): table, keyboard, hand, partial, and clutter.

iPhone 5 and *S*amsung Galaxy S3 (GT-I9300). Examples of frames with each condition are presented in Figure 24.4.

Thus, 500 videos were produced: (50 documents) × (5 conditions) × (2 devices). Each video was at least 3 seconds in duration and the first 3 seconds of each video were split with 10 frames per second. In total, there are 15000 annotated frame images in the dataset.

For each extracted video frame, we have prepared a document location ground truth. The annotation was performed manually. If the corners of the document are not visible on the frame, the corresponding coordinates' point is extrapolated outside the frame (if the document is not visible on the frame at all, all four points will lay outside the frame boundaries).

For each unique document, we have prepared ground truth containing its geometric layout represented by document field quadrangles and UTF-8 string values for text fields. The ground truth is given in JSON format.

In total, in the dataset, there are 48 photo fields, 40 signature fields, and 546 text fields. In addition to fields written using Latin characters with diacritics, the dataset contains fields written using Cyrillic, Greek, Chinese, Japanese, Arabic, and Persian alphabets.

Combined annotation of document boundaries for each frame and of objects coordinates in the document layout allows us to perform studies of object detection and information extraction algorithms and measure their accuracy characteristics and their robustness against various distortions. The latter is especially relevant for performing objective studies of complex document analysis systems, where the result of each processing stage is predicated by the accuracy of its predecessor in the workflow.

Thus, the MIDV-500 dataset has a lot of advantages. However, the variability of capturing conditions in MIDV-500 did not address some of the key issues, mainly significant projective distortions and different lighting conditions. There is a extension of MIDV-500 called MIDV-2019 [23], containing video clips shot with modern high-resolution mobile cameras, with strong projective distortions and with low lighting conditions. In addition, there is a dataset MIDV-LAIT featuring documents with Perso-Arabic, Thai, and Indian scripts [38], the MIDV-2020 dataset with higher variation of text field values [26], and DLC-2021 dataset [105] targeting the problem of document liveness verification.

In this chapter, some results are also presented on several closed datasets containing images of bank cards and Russian identity documents obtained from different sources (such as scans, photos, web cameras, and mobile devices).

24.4 Document Location and Identification

Document image identification (or document image classification) is an assignment of the document image to one of several categories based on its contents. Classification plays a significant role in document recognition process. Specific knowledge about the document type allows us to define optimal parameters for document image

processing in order to achieve the high recognition accuracy. Usually, the problem of document identification is solved in conjunction with the task of document location, which is a task of determining the document boundaries in an input image.

The traditional approach for location and template matching for documents entails an extraction of graphical primitives (static elements) in an input image, after which a correspondence is sought with the primitives defined in a stored template [33]. However, this approach has a significant disadvantage: static elements' search is a hard problem which requires either a non-trivial pre-processing of the document image (which in itself has to be based on the prior knowledge of the document type [133]) or the usage of machine learning methods (which often cannot be applied due to strict performance constraints [67,88]). Besides, the task becomes more complicated considering input images obtained in uncontrolled environment (e.g. if the document is photographed using a small-scale digital camera of a mobile device). This "extended" problem statement does not allow the usage of classical methods based on static elements analysis.

Thus, in mobile document recognition systems, the ideal document classification method should not rely on structural elements' extraction or on OCR results and should only use some basic visual features of the input image. However, due to the high variability of the problem, currently, there is no universal algorithm for document classification which complies to all these requirements and which could classify arbitrary documents in uncontrolled conditions. In this section, we consider some of the methods which could be used to solve this problem, at least in some specific cases, important for practical applications.

24.4.1 *Statistical methods of document identification*

In practice, it can be observed that identity documents often have a fixed geometry (i.e. samples of the same document type are identical if one disregards the document data which have to be extracted). For instance, Internal Russian passport (see Figure 24.5) is one of the documents which has this property. For such cases, the task of template matching can be reformulated in terms of detection of an object with a pre-defined geometry in an image. Such tasks can be



Figure 24.5. An example of Internal Russian passport.

solved using Viola–Jones method [9] which reduces this problem to a binary classification of all subwindows of the image.

The Viola–Jones algorithm [138] is a real-time robust algorithm for object detection, designed to locate faces in digital images. Key features of the algorithm are as follows:

- Haar features similar to those employed in [100] are used as the basic features for classifier construction.
- Use of a so-called integral image allows calculation of basic attributes in an efficient manner.
- AdaBoost [58] as an effective machine learning algorithm is used to construct a strong classifier.
- A cascade structure of classifiers is used to decrease the time required for image processing.

The document classification algorithm proceeds in a similar manner to the Viola–Jones algorithm yet with certain adaptations. First of all, the feature space should be reduced. In our experiments, we still employ Haar patterns whose features consist of two classes: vertically and horizontally oriented patterns, consisting of differently sized rectangles. This is because of the inherent differences in structure of image documents in comparison with face images. For example, the horizontal “triplet” feature of the Viola–Jones classifier practically

Table 24.3. Classifier learning statistics.

ID type	Scans	Ph. copies	Scaled size	Core size
Internal passport	673	527	48×64	50
Foreign passport	100	12	46×64	35
Driving license	91	90	32×20	35
Pension insurance 1	423	3	32×20	35
Pension insurance 2	380	50	38×26	50

provides nose detection. On the other hand, most identity documents have no contrast elements of predefined width. Rather, text blocks often have left (or top) borders fixed, while the opposite ones are floating. In this case, simple "duplet" features are more effective.

As usual, there is no need to use of a cascading scheme for the classifier. As we have determined experimentally, for classification of documents, the detector contains no more than 50 neurons (weak classifiers). This allows for sufficient speed of the detection algorithm even without cascades.

In practice, the size of the document is often smaller than the whole image. In this case, to obtain the placement of the document in the image, the trained classifier is scanned across different locations. Additionally, the classifier can be easily rotated by 90°, 180°, and 270°. By scanning across the image using these rotated classifiers, it is possible to detect the rough orientation of the document.

Experimental results: The following document types have been recognized using supposed algorithm: internal passport for Russian citizens, foreign passport for Russian citizens, Russian driving license, and two types of pension insurance certificates. Examples of these are given in Figure 24.6.

To train each classifier, sets of positive and negative examples have been prepared. Each set of positive examples comprises a mixture of gray-scale scans of the original documents and black-and-white photocopies. Thus, the derived classifiers are able to successfully detect both original documents and photocopies. The sets of negative samples are derived from the other image classes. Detailed information about the training data of the classifiers of all documents types is presented in Table 24.3.



(a) (b)

(c) (d)

(e)

(f)

Figure 24.6. Document types: (a) internal passport; (b) internal passport (photocopy); (c) foreign passport; (d) driving license; (e) pension insurance certificate type 1; (f) pension insurance certificate type 2.



(a) (b) (c)

Figure 24.7. Classifier training for Russian internal passport: (a) downscaled positive sample; (b) mean image of all positive samples; (c) information density map of trained classifier.

The Viola–Jones algorithm does not require high-resolution images, while appropriate downscaling of document images allows for high-speed processing. Clearly, the more we downscale, the higher the processing speed, but at the same time, detection quality might deteriorate. Therefore, it is important to choose an appropriate scale. In our approach, we chose the scale as that scale at which the different document types were still found as distinguishable by human observers.

The training stage for classifying documents proceeds in the following steps:

(1) Compute an integral image

$$\hat{I}(x,y) = \sum_{x' \le x, y' \le y} I_i(x,y) \tag{24.1}$$

for each image I_i and initialize the weights.
(2) Select the best weak classifier *BestWC*.
(3) Add *BestWC* to the strong classifier *SC*.
(4) Evaluate *SC*. If it correctly evaluates all positive and negative training samples, exit.
(5) Update example weights and go to Step 2.

One way to estimate the quality of the positive sample set is to analyze the image obtained by averaging of all positive samples, as shown in Figure 24.7(b). We can note that the features of Russian citizen passports are traced onto this average image.

Table 24.4. Classification results on unseen documents, presented in terms of a confusion matrix: actual documents from top to bottom, classified documents from left to right.

	IP	FP	DL	PI1	PI2	OD
IP	153	0	0	0	0	2
FP	0	27	0	0	0	2
DL	0	0	31	0	0	2
PI1	0	0	0	67	0	8
PI2	0	0	0	0	12	2
OD	3	1	1	6	8	291

Notes: IP = internal passport; FP = foreign passport; DL = driving licence; PI1 = pension insurance form type 1; PI2 = pension insurance form type 2; OD = other documents.

Knowledge about geometry and placement of elementary features can be obtained with a so-called information density map, shown in Figure 24.7(c). Here, a region in the image is darker when more features cover it. More precisely, the brightness of a pixel is indirectly proportional to the summed weight of all features covering the pixel. For example, we can note that many features are concentrated in the region of the photo and stamps.

Once the classifiers are successfully trained, classification, that is, document identification, is straightforward and is performed following:

(1) Compute the integral image for the document image using equation (24.1).
(2) Using SC, evaluate each part of the image.
(3) Merge detections: Two detections are merged if they overlap with 75% or more.

For evaluation of the learned classifiers, a set of (previously unseen) 324 photocopy scans and 276 scans of original documents of physical size A4 were used. The classification results for this test set are given in Table 24.4.

From Table 24.4, we can see that the proposed approach to document image classification affords excellent recognition performance.

Despite the similarity between the different types of identity documents, not a single identity document was mistaken as belonging to another class of ID documents.

24.4.2 *Geometric distortions' correction*

Projective distortions which are significant for Viola–Jones classifier can be neutralized using preliminary assessment of document corners' position in the frame or the position of the camera in relation to the frame.

Quadrangle search: The task of searching the document in the frame does not require the presence of text and can only rely on the differences between the document and background. In addition, the document detection can be regarded as the first step in a series of document recognition methods.

Complications which have to be taken into account when construction such algorithms include the following:

- The document boundaries may not be straight, they can have ruptures or sharp bends.
- The boundaries can have low contrast.
- The document can be partially indistinguishable from its background.
- Lighting conditions are not controlled.
- Lighting may be insufficient and unbalanced.
- The document can have complex structure (e.g. contain a border parallel to the boundaries).
- The background can also have a complex structure (e.g. capturing of the document can be performed against a "chess board").
- The document can be partially outside the frame boundaries.

Construction of boundaries map: The search of document boundaries is usually performed with the assumption of them being orthotropic, i.e. predominantly oriented along coordinate axes. In this case, the "vertical" and "horizontal" boundaries can be detected independently. Let us independently compute the derivative images after which we determine the noise coefficient for each boundary pixel — the ratio of the sum of distances to the closest boundary pixels along the axis (i.e. in case of horizontal boundaries — to the top and

bottom one) to the height of the zone. The background part, containing such type of noise, is usually located only to the one side of the document boundary, thus such coefficient will be higher on average for a true boundary than for false boundaries appearing in the background. If now the intensities of pixels are multiplied by the acquired coefficients, it will suppress the neighboring false boundaries [120].

Let us now unite the boundary pixels in connected components for further filtering. The orthotropy allows us to perform this with a single pass through the image along the considered axis. Acquired components allow for a wider variety of pruning criteria than single pixels, for instance, the minimal, average and maximal intensities, length, etc. Thus, we can obtain either a gray-scale map of the boundaries, retaining the intensities of pixels or components, or a binary one.

Lines and segments search: The segments can be extracted using contours analysis. The method described above yields 6-connected contours. The segment can be defined as a set of consequent contour points such that in the neighborhood of each point, a hypothesis of contour straightness is satisfied. In order to test this hypothesis for each point of the contour, we can estimate a covariance matrix for pixels in the neighborhood with a fixed size l. Eigenvalues and eigenvectors of this matrix allow us to estimate the size and shape of the intensity distribution on a part of the contour, by approximating it with an ellipse. The straightness hypothesis is considered true if the ratio of the axes of this ellipse is higher than a pre-defined threshold.

Another method for straight lines and segments extraction is an application of the Hough transform [17]. Let us use this iterative procedure [120]:

(1) Compute Hough transform of the boundaries map.
(2) Select a line which corresponds to the global maximum in Hough space.
(3) Erase the neighborhood of this line from the boundary map.
(4) If the number of obtained candidates is smaller than required, return to Step 1.

The procedure described above will not generate false lines from local maxima in Hough space. The value in Hough space can be used as an estimation of the line in the following processing methods. The classical Hough transform computation scheme is not efficient

in practical recognition systems due to high computational complexity. It is recommended to use the fast Hough transform [98] which requires $4n^2 \log n$ addition operations for an image with $n \times n$ pixels. This algorithm independently processes each of the four 45-degree ranges. By analogy with "vertical" and "horizontal" boundaries, we will independently find "vertical" and "horizontal" lines in ranges $[45°, 135°]$ and $[-45°, 45°]$, respectively.

Quadrangle variants construction: Consider two ways of determining document corners in the image using the obtained segments and lines. In case where there are only lines we can construct all possible quadrangles using these lines. The base weight of each such quadrangle is a sum of weights of lines used in its construction. Since in a general case the lines should not continue after the quadrangle vertices, a penalty can be computed for each corner as a sum of neighboring pixel intensities along the lines outside the quadrangle.

In order to construct a quadrangle using found segments, a graph of possible quadrangle sides can be used. Vertices of such graph correspond to the found segments, and edges to the intersection points of the lines containing the segments. In order to build the graph, all segments are separated to predominantly vertical and predominantly horizontal ones, depending on the angles between the segments and coordinate axes of the image. If the segment can be grouped with both classes (i.e. it is a "diagonal" segment), the segment is added to both groups. A weight is associated with each vertex of the graph, equal to the weight of the corresponding segment. Edges of the graph are constructed as intersections of all horizontal segments with all vertical segments. If we consider that the graph is constructed for N segments, the graph contains $O(N^2)$ edges in total. Each edge is classified according to the document corner type (left-top LT, top-right TR, right-bottom RB, and bottom-left BL). The classification is performed according to the position of intersection point relative to the adjacent segments. If the intersection point lies inside both segments (i.e. if it is the true intersection point of the segments and not only of the lines which contain them), it can correspond to an arbitrary corner ($LTRB$); if it is positioned to the left of the horizontal segment and inside the vertical segment, it can be left-top or bottom-left (BLT). A weight is associated with each graph edge, which is equal to the minimal distance between the ends of the segments which produced the corresponding intersection point. The graph constructed

in such way can be expanded as a four-partitioned directed graph such that each edge has an unambiguously defined type. Each 4-edge cycle in this graph, which passes through four different vertices, corresponds to a candidate quadrangle in an input image. Finding such cycles can be performed using depth-first searches with depth limited by 4, which start from one of the graph partitions and may enter the same vertex multiple times. A weight is associated with each quadrangle, equal to the sum of weights of all edges and vertices in the corresponding cycle.

Projective model of a rectangle: While constructing the quadrangle, we did not take into account that it had been a projection of a rectangular document (possibly even with known aspect ratio) on the image plane. If we assume that the document is rectangular from the start, it becomes possible to determine whether the found quadrangle is a feasible projection. In order to do this, let us consider a centrally symmetrical model of image construction — tetrahedral angle with vertex O in the center of the camera lens, and with section $A'B'C'D'$, produced with the image plane, which intersects with the angle edges in the corners of the found quadrangle. It is known that every tetrahedral angle has only one (within the set of parallel displacements) section $ABCD$ in the shape of parallelogram. Vertices of such section (with $OD = OD'$) can be computed as [147]

$$\begin{aligned}
\overrightarrow{OA} &= \overrightarrow{OA'} \cdot (\vec{b} \wedge \vec{c} + \vec{b} \wedge \vec{d} + \vec{c} \wedge \vec{d}) \;/\; k, \\
\overrightarrow{OB} &= \overrightarrow{OB'} \cdot (\vec{a} \wedge \vec{c} - \vec{a} \wedge \vec{d} + \vec{c} \wedge \vec{d}) \;/\; k, \\
\overrightarrow{OC} &= \overrightarrow{OC'} \cdot (\vec{a} \wedge \vec{b} - \vec{a} \wedge \vec{d} + \vec{b} \wedge \vec{d}) \;/\; k, \\
k &= \vec{a} \wedge \vec{b} - \vec{a} \wedge \vec{c} + \vec{b} \wedge \vec{c},
\end{aligned} \tag{24.2}$$

where $\wedge$ is the operation of the skew products of vectors, $\vec{a}, \vec{b}, \vec{c}, \vec{d}$ are two-dimensional vectors $\overrightarrow{O'A}, \overrightarrow{O'B}, \overrightarrow{O'C}, \overrightarrow{O'D}$, and O' is the projection of O on the image plane.

The restriction of the angle ranges and aspect ratios of the obtained parallelograms allows us to disregard quadrangles which do not comply to the assumed conditions [119].

Thus, when the document quadrangle is found, we can calculate the current projective transform and correct the projective distortion

of the document to apply the methods described above. However, the detection of document quadrangle is not always a simple task. There are some document types that do not contain clear borders, therefore document quadrangle detection is not applicable. For such cases, the problem of calculation of projective transform can be solved by finding line bundles and vanish points [74, 114].

24.4.3 *Template matching*

It is quite hard to obtain enough real data for statistical and machine learning methods. In addition to methods of augmentation [91] and synthesis of training data, there exists an approach to classification based on template matching [118]. In case of document image, a single high-resolution document image, obtained using scanner and cropped by the document boundaries, can be used as a matching target. An important step is to construct a representation of an image with which, on the one hand, the matching would be robust against the distortions described above and would provide high classification accuracy, and on the other hand, with expansion of the number of classes, the complexity of the task would not significantly increase. Image representation in metric space allows us to use distance functions (such as Hamming or Euclid distance) as error function and efficient data structures for nearest neighbors' search (such as FLANN [94]). Unfortunately, today, there are no proposed methods for metrical representation of the full image suitable for our tasks. However, the methods for constructing metric representations of small image fragments, which provide high classification quality, are known. The task considered in the latter case is easier, as the issue of extraneous objects is eliminated, and projective distortions can be approximated by affine ones or even by similarity transforms. Robust search for the most informative image fragments — "keypoints" — allows us to represent an image as a set of local metric descriptors. The "bag of features" model is not in itself sufficiently robust against cross-class collisions and does not provide precise document location. However, the classification stability and location precision can be improved using "features constellation" model, i.e. by accounting for the information about relative position of keypoints for estimating geometric correspondence between the representations. The general mapping to the metric space for the "features constellation" model

is unknown, thus the correspondence algorithm consists of two steps. The first step runs an approximate nearest neighbors search in a "bag of features" model. The second step independently estimates the geometric correspondence with each candidate selected on the first step.

Compact representation: In the main case for one document class, there is a single ideal target. Document zones can be roughly divided into static elements (background features, logos, titles of the document, and its fields) and the data (values of text fields, table cells, personal photo, etc.). The latter zones, as a rule, cannot be correctly matched with the ideal target and thus introduce negative effect to the algorithm performance. On the ideal target, such zones should be preliminarily selected and the keypoints found in such zones should be excluded. In cases which have several targets in paper [11], it is proposed to use all available keypoints without analyzing the document zones. The selection of keypoints' detection algorithm and the methods of their metric description remains an open question. The main requirements are robust detection, stability and uniqueness of the descriptors, and computational performance. Preliminary normalization of the input image (such as smoothing, scaling to the standard DPI, and color reduction) in combination to the scale pyramid allows us to use algorithms which are less robust in a general case but which can be computed much quicker (such as binary descriptors RFD [51] and BRIEF [30]). It is worth noting that among the set of ideal targets, there could be images more rich with local features, and thus it could be wise to restrict the number of keypoints on each level of the scale pyramid for each target. In general, the image I is represented as

$$W(I) = \{\langle p_i, f_i \rangle\}_{i \in [1,M]}, \tag{24.3}$$

where $p_i = \langle x_i, y_i \rangle$ are the coordinates of i-th keypoint, f_i is a descriptor of the i-th keypoint neighborhood, and M is the number of keypoints.

Geometric verification: Let H be a transform which converts the points of the image I to the points of the target T within a pre-defined margin of geometric error δ. H can be estimated using the family of RANSAC [56] methods. The pair of points $\langle p, p' \rangle$ with similar

descriptors is considered a valid correspondence if:

$$|H(p) - p'| < \delta, \quad p \in W(I), \quad p' \in W(T). \tag{24.4}$$

As the final result such target T_b is selected for which the number of valid correspondences $G(W(I), W(T_b))$ is maximal. If their number is smaller than a pre-defined threshold, the input is rejected in order to minimize the amount of false matchings. In the case when two candidates have the same number of correspondences, an additional transform estimation is computed:

$$e_i = \frac{1}{G(W(I), W(T_i))} \sum_{j=0}^{G(W(I),W(T_i))} \frac{|H(p_j) - p_j^i|}{\delta}, \quad \delta > 0, \tag{24.5}$$

$$p_j \in W(I), \quad p_j^i \in W(T_i).$$

Document image classification is an important task in automated document processing applications. In this section, we have presented a fast and accurate method for classifying form images based on visual appearance and without the need for OCR, document segmentation, or any form of pre-processing. As has been shown, our approach performs well in classifying different types of identity documents and should be equally applicable to other document types, such as application forms or questionnaires.

24.5 Text Field Location

Location of text fields is one of the essential parts of the document recognition process. Document text field location method is responsible for searching field boundaries and determining the correspondence between fields and document attributes. The final document recognition result strongly depends on how well the text field locations are found. The task is quite difficult, there are several approaches dedicated to it [68], and there is no universally accepted one. The selection of text field location methods is conditioned by the initial problem statement, the characteristics of input data, and various constraints.

Consider the problem statement for document text field boundaries' location. The document initially consists of one or several



Figure 24.8. Text fields of the data zone of the third page of Russian passport.

zones, each containing text fields which compose the content of the document. The zone itself consists of text lines, each of which contains one or multiple text fields. Fields in a given text line are separated with some distance, characteristic to the given document type, and the text blocks which are closer together in relation to this distance are considered parts of a single text field. The number of lines and fields in a document can vary. The task is to determine the boundaries of such fields and match them to the corresponding attributes of the document.

For example, for the data zone of the third page of Internal Russian passport (Figure 24.8), it is required to determine positions of the following fields: last name, first name, patronymic, gender, birth date, and birth place. The "patronymic" field may be absent, and the number of lines of the "birth place" field vary from one to three (such fields are highlighted in Figure 24.8 with a lighter border).

In order for the algorithm to successfully find text fields, it should comply to the following criteria:

- robustness to zone location errors,
- robustness to the changes in lighting,
- robustness to the document background,
- high processing speed,
- the ability to estimate the confidence of the result.

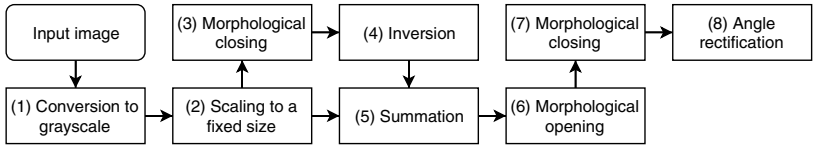


Figure 24.9. Image pre-processing algorithm.

The algorithms presented in this section consist of two major steps: preliminary processing of the input image and the extraction of text lines and fields.

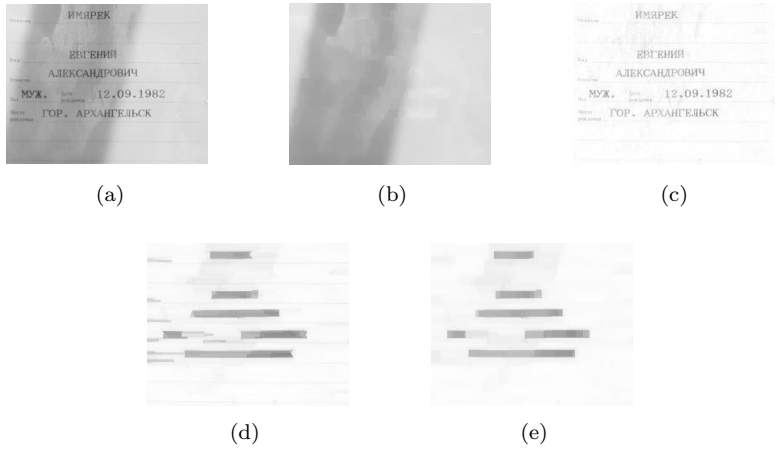
24.5.1 *Image pre-processing*

The approach to image pre-processing largely involves morphological operations. This approach has already been used for text lines location, e.g. in [47, 142], however, the particular processing scheme proposed here is concentrated specifically on text field location taking into account special characteristics of images captured using mobile devices.

The main idea of image pre-processing resides in transforming the input image in such a way that the text fields are reduced to easily detectable and visually identifiable objects. Figure 24.9 presents the scheme of the proposed pre-processing algorithm.

Main morphological operations used for image pre-processing are dilation and erosion, which are performed for each pixel of an image by computing max and min values respectively in its neighborhood, defined as some geometric primitive. Derivative operations are morphological *closing*, which smooths object contours, breaks thin "bridges", and eliminates "protrusions", and *opening* which fills contour gaps and merges adjacent objects (Figure 24.10).

The computational complexity of morphological operations linearly depends on the size of the neighborhood primitive, thus morphology can be computationally demanding. However, there exists a van Herk algorithm [135] which allows us to compute morphological operations with rectangular primitive, without the computational complexity depending on the size of this rectangle. In order to further speed-up this algorithm, it is possible to employ SIMD commands, such as SSE command set on x86 processors or NEON on ARM processors, which is especially relevant for mobile devices [85].



(a) (b) (c)

(d) (e)

Figure 24.10. Image pre-processing results visualized step by step: (a) scaling (2); (b) closing (3); (c) summation (5); (d) opening (6); (e) closing (7).

24.5.2 *Text field location using image processing methods*

The first proposed method for text field search is based on the selection of text lines and the following matching of the found objects to the document template. The method takes as an input the image pre-processed according to the previous step.

Selection of text lines and fields:

(1) The input image contains characteristic components corresponding to text fields. In order to separate them from the background, we compute the threshold value using one of the methods of binarization, such as Otsu [99]. This threshold value is then used to compute histograms on the following steps of the algorithm.
(2) Vertical histogram is computed by summation of only those pixel intensity values which are lower than the computed threshold since text components are taken to be darker than the background (see Figure 24.11).
(3) Upper and lower boundaries of text lines are computed by finding peaks in the histogram. In order to eliminate outliers, such characteristics can be used as minimal and maximal character heights. The peaks with size significantly smaller than the minimal character height can be interpreted as outliers. If the size of



Figure 24.11. Calculation of vertical histogram.



Figure 24.12. Calculation of horizontal histogram.

the peak is larger than the maximal character height, this signifies an aggregation of the neighboring text lines and an additional analysis is required in order to find the appropriate horizontal cuts.

(4) For each found text line, a horizontal histogram is computed using the threshold, similar to Step (2).

(5) Text field boundaries inside text lines are found by analyzing the horizontal histograms (see Figure 24.12). In order to eliminate outliers, the minimal allowed width of the text field can be used (e.g. equal to the minimal width of a single character). In order to separate different text fields inside one text line, a characteristic could be introduced, such as the minimal distance between fields. This is justified, since in the majority of identity documents in order to visually distinguish the text fields containing different data, they are usually separated with a distance wider than one character. Thus, components which are closer together are merged into a single field, and distant components will remain as separate fields.

(6) Morphological operations can partially deteriorate elements of the first and last characters of the text fields, as well as eliminate subscripts or diacritics. In order to mitigate that the found text fields, boundaries are widened horizontally and vertically.

Matching of the found text fields to the document zone template: The method presented above outputs a set of text line

coordinates and coordinates of found fields, however it is also necessary to match the found text fields to the document attributes, accounting for possibly absent fields. For instance, for the data zone of Internal Russian passport, the field "patronymic" may be absent, and the number of lines corresponding to the "birth place" field varies from one to three. In order to solve this problem, it is proposed to estimate the correspondence of the produced set of lines and fields to the document zone template.

In order to describe the zone template, we can use the geometrical model of relationships between lines and fields. Let us consider the document zone as a set of lines and each line as a set of fields. For text lines, a relationship "above"/"below" is defined, and for fields, "to the left"/"to the right". The text lines and fields can be marked as optional and they can have additional properties, such as characteristic sizes, position inside the document zone (e.g. the text field must be pinned to the left border), and others, and each of them is used for estimating the correspondence of the obtained data to the template. After that, an exhaustive search is performed among all possible correspondence variants, computing the estimation of each variant.

If there are no correspondence variants or they all have very low estimations, the document zone is rejected, and if there are several viable candidates, the single variant is selected with the largest estimation value or additional selection problem is considered. The rejection in most cases signifies that the document boundaries were found incorrectly or the input image had insufficient quality.

Experimental results: Consider the results of the presented algorithm on the set of images of Internal Russian passports, obtained from different sources: scans, photos, web camera frames, and video frames from a mobile device. Some images are black-and-white copies with poor quality. The results are presented in Table 24.5.

The presented algorithm of text field location shows its applicability and decent quality. Important advantage of this algorithm is high versatility and computational performance, as well as robustness to document background, lighting, and document zone distortions. This algorithm can be used to perform per-field segmentation for more than 1000 different documents from all around the world (see Figure 24.13).

Table 24.5. Results on the set of Internal Russian passport images.

	Images in total	Document boundaries detection	Found text fields
Quantity	2796	2711	2675
Percentage	100.00%	96.96%	95.67%



Figure 24.13. Text field location results on different document types.

24.5.3 *Dynamic programming approach to template-based recognition*

Template-based recognition is an approach which involves recognition of composite objects on an image with the use of separate parts models and joint geometric relations models described by a template. As a result, correspondences between model parts and image regions are built. Location of the structured document's text fields is also a problem of template-based recognition.

The problem of template-based recognition can be stated in a certain manner, so dynamic programming can be used [52]. In particular, it is possible when parts of a structure form a sequence:

$$Z(l) \equiv \sum_{i=1}^{N} m_i(l_i) + \sum_{i=1}^{N-1} d_i(l_i, l_{i+1}) \to \min_{l_1,\ldots,l_N}, \tag{24.6}$$

where $Z(l)$ is a function being minimized, $l = l_1, \ldots, l_N$ are parts locations, $m_i(l_i)$ is a cost for assigning location l_i to i-th part, and $d_i(l_i, l_{i+1})$ is a cost for deformation of a link between adjacent parts.

The general algorithm proposed in [52] has high computational complexity of $O(NW^2)$, where N is a number of parts and W is a number of feasible locations for one part. We propose a more efficient $O(NW)$ algorithm named Dynamic Squeezeboxes Packing (DSP). To allow the use of this algorithm, we require specific form of cost functions $d_{ij}(l_i, l_j)$. We investigate DSP use for the OCR problem of text fields' extraction.

Dynamic squeezeboxes packing algorithm: Let us specify the optimization problem (24.6) with costs for link deformations in the following form:

$$d_i(l_i, l_{i+1}) = \begin{cases} 0, & T_{\min}(i) \leqslant l_{i+1} - l_i \leqslant T_{\max}(i), \\ +\infty, & \text{otherwise}, \end{cases} \tag{24.7}$$

where $T_{\min}(i)$ and $T_{\max}(i)$ are distance thresholds between ith and $(i+1)$th parts. In this case, the function (24.6) can be simplified if constrains are added:

$$\begin{gathered} Z(l) = \sum_{i=1}^{N} m_i(l_i) \to \min_{l}, \\ T_{\min}(i) \leqslant l_{i+1} - l_i \leqslant T_{\max}(i), \quad i = 1, \ldots, N-1. \end{gathered} \tag{24.8}$$

Such distance variations remind squeezebox playing, hence the name.

The deformation model (24.7) makes an investigated object impossible to recognize correctly in case of deformations exceeding the boundaries. All the deformations within the boundaries are equally feasible and not penalized.

Text fields extraction: Input of the visual inspection zone (VIZ) segmentation algorithm consists of gray-scale normalized image of VIZ $I(x, y)$ (as before) and a template with *a priori* information about zone structure. The template is represented by top-down sequence $\{\phi_i\}$ of rows one of two types: "text string" and "gap" alternating with each other. For each row ϕ_i, the minimum $h_i^{\min}$ and maximum $h_i^{\max}$ heights are specified. Besides that, each row of type "text string" has its own structure — sequence of alternating blocks $\{\psi_{i,j}\}$ of two types: "field" and "gap". For every block $\psi_{i,j}$, width limits $w_{i,j}^{\min}$ and $w_{i,j}^{\max}$ are known.

The output of the algorithm is expected to be a set $L = \left\{t_i, b_i, \{l_{i,j}, r_{i,j}\}_{j=1}^{M_i}\right\}_{i=1}^{N}$ of upper and lower borders $[t_i, b_i)$ for every horizontal row ϕ_i as well as left and right borders $[l_{i,j}, r_{i,j})$ for each block $\psi_{i,j}$ so that blocks of type "field" cover physical text with minimum area of background. It is required that the rows must not intersect and must cover the image completely.

Let us assume that the average brightness of the text in the document image is less than the average background brightness. The more text and the less background are inside the fields, the better the segmentation result. Therefore, for the optimization problem, it is natural to choose a function, which is optimal at high average brightness of pixels outside the fields and low one inside. An example of such function is inter-class dispersion [99], multiplied by the sign of the difference in the average brightness of the classes:

$$V = \omega_0\omega_1(\mu_0 - \mu_1)|\mu_0 - \mu_1| \to \max, \tag{24.9}$$

where ω_0 and ω_1 are fractions of the zero and the first classes from all the image pixels, μ_0 and μ_1, are mean intra-class brightness values. The pixels inside the "gaps" belong to the zero class. The pixels inside the "fields" belong to the first class. To increase the inter-class contrast, we apply a number of image processing operations, such as additional morphological processing and autocontrast.

The function (24.9) cannot be represented as a sum of single part costs. However, if we assume that the row heights h_{2k} and the field widths $w_{2k,2m}$ are fixed, then the maximization of V is equivalent to minimizing the total brightness of the pixels within the "fields".

Thus, we obtain the following optimization task:

$$S_1 = \sum_{k=1}^{\lfloor N/2 \rfloor} \sum_{m=1}^{\lfloor M_{2k}/2 \rfloor} \sum_{x=x_{2k,2m}}^{x_{2k,2m}+w_{2k,2m}} \sum_{y=y_{2k}}^{y_{2k}+h_{2k}} I(x,y) = \sum_{k=1}^{\lfloor N/2 \rfloor} \sum_{m=1}^{\lfloor M_{2k}/2 \rfloor} I_\Sigma(x_{2k,2m}, y_{2k}, w_{2k,2m}, h_{2k}) \to \min, \tag{24.10}$$

where $I_\Sigma(x,y,w,h)$ is the total brightness in the window with the upper left corner at the point (x,y), width w and height h. Note that its value can be computed for any valid values of the arguments for the constant number of operations if we first calculate the integral image [137] for $O(WH)$ operations.

Suppose that for any position y of kth row the minimum total brightness of its fields is known:

$$S_k(y) = \min_x \sum_{m=1}^{\lfloor M_{2k}/2 \rfloor} I_\Sigma(x_{2k,2m}, y, w_{2k,2m}, h_{2k}). \tag{24.11}$$

Then, problem (24.10) reduces to

$$\sum_{k=1}^{\lfloor N/2 \rfloor} S_k(y_{2k}) \to \min_y. \tag{24.12}$$

Taking into account the constraints for y values,

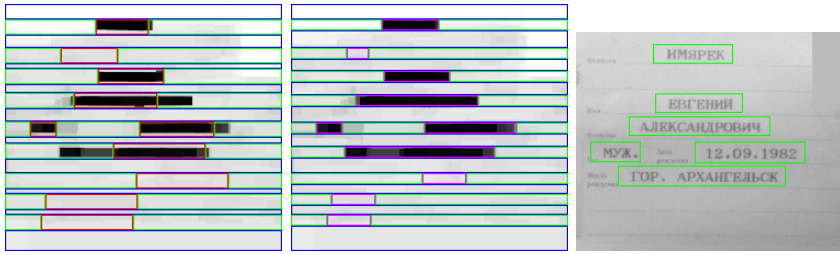
$$h_{2k} + h_{2k+1}^{\min} \leqslant y_{2k+2} - y_{2k} \leqslant h_{2k} + h_{2k+1}^{\max}, \quad k = 1, \ldots, \left\lfloor \frac{N-1}{2} \right\rfloor. \tag{24.13}$$

Thus, the task meets form (24.8) that allows us to apply DSP algorithm to solve it. Restrictions on the positions of the first and last rows can be set by infinite penalties:

$$S'_2(y) = \begin{cases} S_2(y), & h_1^{\min} \leqslant y \leqslant h_1^{\max}, \\ +\infty, & \text{else}; \end{cases} \tag{24.14}$$

$$S'_{N-1}(y) = \begin{cases} S_{N-1}(y), & H - h_N^{\max} \leqslant y + h_{N-1} \leqslant H - h_N^{\min}, \\ +\infty, & \text{else}. \end{cases}$$

Problem (24.11) of calculating the optimal segmentation of fields within a single row in a current position reduces to problem (24.8) in



(a) (b) (c)

Figure 24.14. Stages of VIZ segmentation: (a) result of minimizing S_1 with static sizes of fields using DSP; (b) result of maximizing (24.9) using coordinate descent; (c) result of VIZ segmentation.

a similar way. Hence, problem (24.10) of segmenting rows and fields of fixed sizes can be solved by DSP in two stages: search for the optimal position of the fields for each possible position of each row and then search for the optimal position of the lines themselves (see Figure 24.14(a)).

Let us recall that according to the initial problem statement, the sizes of the fields can vary within the boundaries set by the template. To refine the size and position of the fields, it is proposed to maximize (24.9) using coordinate descent: alternately looking over the fields and their boundaries within template boundaries, increasing the target value of the function. As an initial approximation, we use the dynamic programming solution. The stopping criterion will be considered as idle iteration search without increasing the value of the function, or reaching the maximal number of iterations given as a parameter of the algorithm (see Figure 24.14(b)). Final extracted fields are represented in Figure 24.14(c).

The total complexity of the algorithm is $O(WH\sum_{k=1}^{\lfloor N/2\rfloor} M_{2k})$ (the complexity of the integral image is $O(WH)$, dynamic programming within rows is $O(WH\sum_{k=1}^{\lfloor N/2\rfloor} M_{2k})$, dynamic programming on the rows is $O(HN)$, and coordinate descent is $O((W+H)\sum_{k=1}^{\lfloor N/2\rfloor} M_{2k})$).

Experimental results: An application of DSP algorithm was implemented into industrial recognition system: the module for simplifying entry of identity document data Smart IDReader [19].

VIZ fields' extraction was performed on a set of 7700 Internal Russian passports photos. Each photo contained two pages: the one

Table 24.6. Text recognition quality for different fields.

Field	Slugin *et al.* (%)	DSP, 1 step (%)
Authority	55.34	55.37
Authority code	94.10	94.48
Birthdate	90.92	93.09
Birthplace	68.62	68.70
Gender	93.57	95.58
Issue date	92.63	93.63
First name	91.96	94.29
Patronymic	90.94	93.36
Second name	87.34	89.49

with issuing information and the one with photo, personal information, and machine-readable zone (MRZ). Fields' extraction was run on an each page separately. We compared our approach with the template-free one, based on histogram analysis [122]. With the use of our algorithm, the number of incorrectly recognized passports decreased by 12% from 1094 to 958. The number of errors caused by wrong VIZ segmentation was reduced by 38% from 476 to 297. The difference in numbers can be explained with the new errors produced by the following subsystems after the segmentation errors were fixed. In relation to the total number of errors, the share of segmentation errors went down from 44% to 31%.

Table 24.6 shows comparison of recognition quality for different fields between the histogram-based method and our approach with the different number of refinement steps.

24.6 Character Segmentation and OCR

Optical recognition of different objects, such as printed or handwritten letters, is one of the most important tasks in computer vision. In the last decades, a vast variety of methods for object recognition was developed. Recognition systems can be divided into several types: trainable, with unsupervised training, logical, probabilistic, and others [61].

Artificial neural networks (ANN) were initially created to model biological systems, but nowadays, they are very popular and powerful

instrument for object classification. In some cases, neural networks can even compete with human ability to recognize images [126]. As a primary motivation, neural networks use a simplified model of the brain [121]. Modern research and development of this field target improvement of the neural network architectures and training methods. Convolutional neural networks were a breakthrough. Today, almost all used neural networks have convolutional layers.

Convolutional neural network (CNN) was first introduced by Yann LeCun [80]. It has two convolutional layers, two pooling layers, and several fully connected layers to form the final result. The answer of the CNN with this architecture is invariant to small shifts of the input image and extracted features are the same for every part of the input image (local features). A lot of modern CNNs are based on this architecture, for example, AlexNet [76], which also suggests to use several GPU units in parallel for training; ZFNet [145], which introduced method for neural network visualization; VGGNet [117], which shows that depth of the neural network is a very important parameter; and others [63, 107, 125].

Usage of the CNNs in mobile recognition systems can entail some limitations related to performance, since their complexity is higher in comparison with classical fully connected neural networks. To use such neural networks in real-time mobile solutions, we need to develop special methods for extremely high-performance execution of the convolutional layers.

24.6.1 *Distinguishing text from the background*

In the problem of text recognition, the situation when it is not so simple to distinguish text from background is quite common. In case of complex background, the space symbol cannot be defined as "nothing". This problem is especially important in such tasks as bank card recognition, since there are absolutely no restrictions to the background, as shown in Figure 24.15.

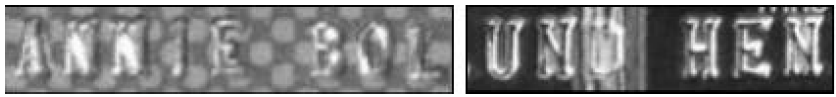


Figure 24.15. Examples of complex backgrounds.

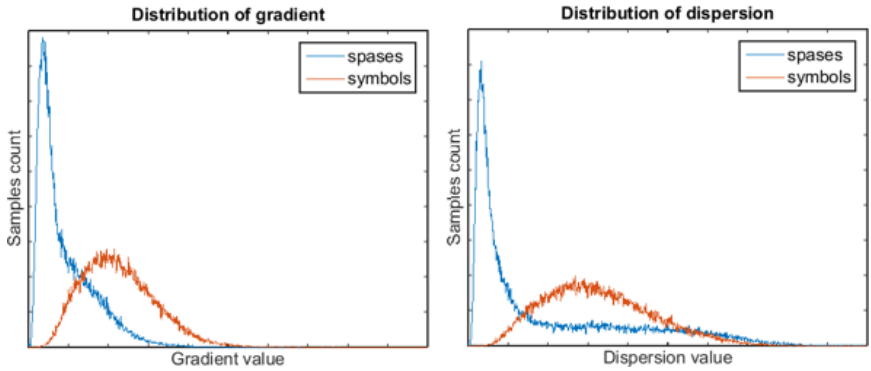


Figure 24.16. Comparison of statistical data for symbols and spaces.

As system's recognition quality increased, it has become completely clear that the task of extracting words from raw symbols of a string is a separate part that requires a solid solution. To be more specific — space symbol detection quality of the base recognition NN is 96.5%, when character recognition quality — 98.7%. At this stage, many problems occur due to background complexity: lines, curves, sparkles, and shades. In order to solve this task, we have to find an efficient and fast way to detect space characters against other symbols.

Standard methods of space detection based on the analysis of image statistics provide unsatisfactory results due to background complexity mentioned above (see Figure 24.16). Along with the possibility of letter to be lighter and darker than background within one symbol, complexities mentioned above do not allow us to use any binarization methods, which are widely used in scanned document recognition [70]. Even modern binarization methods based on convolutional neural networks [14] don't provide sufficient quality in this task.

One of the options to solve this task is to increase the amount of information at the neural network input [113], since even human is not always able to tell for sure whether there is a symbol or a space when looking at a given cropped symbol image, while the answer can be much more confident if a little wider information is given (for example, two neighboring symbols from the string). The main feature of this method is that NN uses both neighboring symbols' images (see Figure 24.17) and the purpose is not to guess whether there is a space

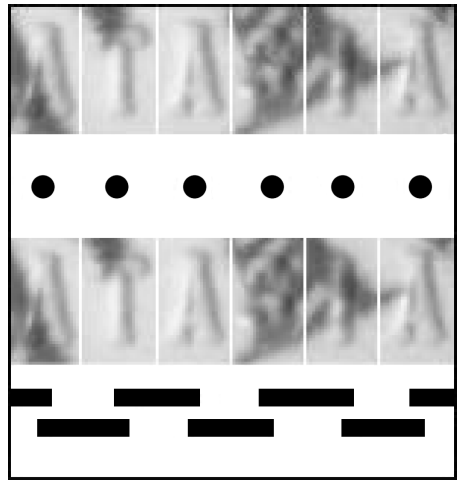


Figure 24.17. Neural networks' applications: classical (top) and suggested (bottom).

or not but to determine a grade of similarity of entities at the parts of input image. This NN has a four-dimensioned output: two spaces, space and symbol, symbol and space, and two symbols.

The described experiment was performed within the development of a recognition system for credit cards. As a reference dataset, we used a set of different images of 4320 cards which provided about 130000 characters. All neural networks were trained using a training dataset of about 230000 images not presented in the reference one. Every image was marked up by a human to retrieve information required for training (in case of the training dataset) and to compare the result (in case of the reference dataset).

We prepared special training dataset consisting of every pair of neighboring symbols with the proper label and used it to train our NN with a four-dimensioned output vector. At the recognition stage, we used paired symbols as input for the NN and compared the result with the result of the main recognition NN that tries to recognize every symbol separately.

Table 24.7 shows that the new NN had better quality in overall and a notable displacement of the error distribution. It had better space detection quality and a slight decrease in symbol detection quality. Application of this NN allowed us to slightly increase the

Table 24.7. Per-symbols' result.

	Spaces	Symbols	Overall
Base NN	93.6%	99.8%	96.5%
New NN	94.3%	99.6%	96.8%
Count	68246	61903	130149

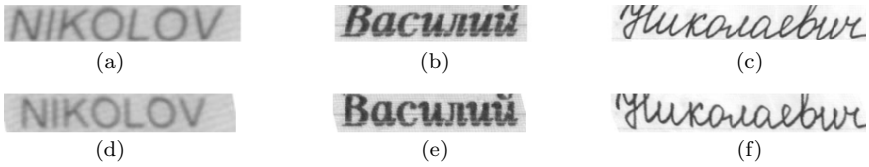
Table 24.8. Errors' correlation.

	Spaces	Symbols	Overall
Errors 1	4392	141	4533
Quality 2	44.7%	29.8%	44.3%
Errors 2	3893	241	4134
Quality 1	37.6%	58.9%	38.9%

recognition quality, but it was not a goal in this case. What is very important here is the errors' distribution, which is different. The first two lines of Table 24.8 represent the amount of errors of the base NN and the quality of the new NN over these samples. The second two lines do the same but for the new and base NNs, respectively. It allows us to combine methods in order to create a new one with better quality. That is how the idea was integrated into the resulting system. We take both estimations and simply make a new one as their weighted combination.

24.6.2 *Italic and slanted writing correction*

After the detection of word boxes, there is a possibility to perform italic/slanted text correction. However, this step is not required in general case, and it is possible to obtain the quality increase in some specific cases. Also, it is possible to simplify the following part of the recognition and, therefore, improve performance, which is very important due to the widely spread mobile devices equipped with high quality cameras. It results in an increased interest or even active revision of the existing approaches for all stages of computationally efficient high-quality document recognition [104].



(a) (b) (c)

(d) (e) (f)

Figure 24.18. Samples of oblique, italic, and handwritten textual blocks: (a) oblique font sample; (b) italic font sample; (c) handwritten block sample; (d) rectification result; (e) rectification result; (f) rectification result.

Oblique, italic, or handwritten textual blocks (or fragments) can be rather widespread in some domains and their successful recognition may become essential for an industrial system (for example, for Internal Russian passports, these blocks are present on at least 3% of all issued documents), which results in a complication of the field processing stage. These font characteristics especially impact the line segmentation into individual characters [34]. The segmentation stage can be simplified by identifying these non-standard cases and restoring them into the "canonical" one (from the point of view of the system designer). For such cases, an application of shear transform to the input image of the textual fragment can be a solution. Real samples of these kinds of blocks and results of their correction using shear transform are presented in Figure 24.18. However, to perform the deslanting procedure, we need to calculate the slant angle of the fragment.

To correct these distortions, we suggest to use methods based on Hough transform [15, 16]. There is an efficient way to calculate Hough transform approximation known as fast Hough transform. Detailed description of its computation is presented in [1, 98]. It differs from the classical transform with another kind of accumulator space parameterization and a reduced complexity of $O(n^2 \log n)$ (as opposed to $O(n^3)$ for the original algorithm). This transform can be efficiently calculated for the following ranges of angles: $[-45°, 0°]$, $[0°, 45°]$, $[45°, 90°]$, and $[90°, 135°]$. Each row of the FHT corresponds to a fixed direction t, on which parallel projections of the image with all possible shifts s are calculated. For every row t, we propose to calculate the sum of squares of intensity derivative:

$$SSG_{DRT}(t) = \sum_{j=1}^{w-1} \left(v_{tj} - v_{t,j-1}\right)^2, \tag{24.15}$$

where w is the length of the row and $v_{i,j}$ is intensity value of the pixel with coordinates i and j, respectively. The choice of this criterion is determined by the presence of high thin peaks on the projection corresponding to the correct direction. The absolute difference of intensities on these peaks achieves large values and squaring them significantly increases their weights in the resulting sum. As was mentioned, FHT and Discrete Radon Transform (DRT) have different coordinate systems, and this fact must be considered for building final criterion. Let"s consider this thesis. For this purpose, let t be the fixed direction. In Radon transform, it corresponds to the angle $\phi_t = \arctan(t/(h-1))$, where h is a height of the source image [17]. Every row of FHT is scaled with multiplier $k_t = 1 \cos \phi_t$ relative to correspondent row in DRT (so if the length of DRT row t equals w, then the row of FHT has length $w \cdot k_t$). At the same time, Radon invariant is preserved for FHT. Thus, it results in the decrease of projection DRT values with the same multiplier. So, we can assume that $\sum_{j=0}^{w \cdot k-1} v_{j/k} \approx k \sum_{j=0}^{w-1} v_j$. FHT criterion calculation in the same row t can be approximated in the following way:

$$SSG_{FHT}(t) = \sum_{j=1}^{w \cdot k_t - 1} \left(\frac{v_{j/k_t}}{k_t} - \frac{v_{j-1/k_t}}{k_t} \right)^2. \tag{24.16}$$

Assuming that $v_{j/k} - v_{(j-1)/k} \approx (v_{j/k} - v_{(j/k)-1})/k$ and using the mentioned approximations, we can derive that

$$SSG_{DRT}(t) = k^3 \cdot SSG_{FHT}(t). \tag{24.17}$$

Thus, we can obtain the approximation of DRT criterion (24.17) by simple multiplication of (24.16) by the value which depends only on the row index t and the height of FHT_{image}.

The difference in criterion values for $SSG_{FHT}(t)$ and $SSG_{DRT}(t)$ is shown in Figure 24.19(d). Row indices are placed on the horizontal axis and criterion values on the vertical axis. Markers represent the coordinates of global maxima. It can be observed that maxima are located in different rows and that the value based on $SSG_{DRT}(t)$ criterion results in correct rectification of textual fragment (see Figure 24.19(c)).

The final algorithm can be described as shown in Algorithm 1.

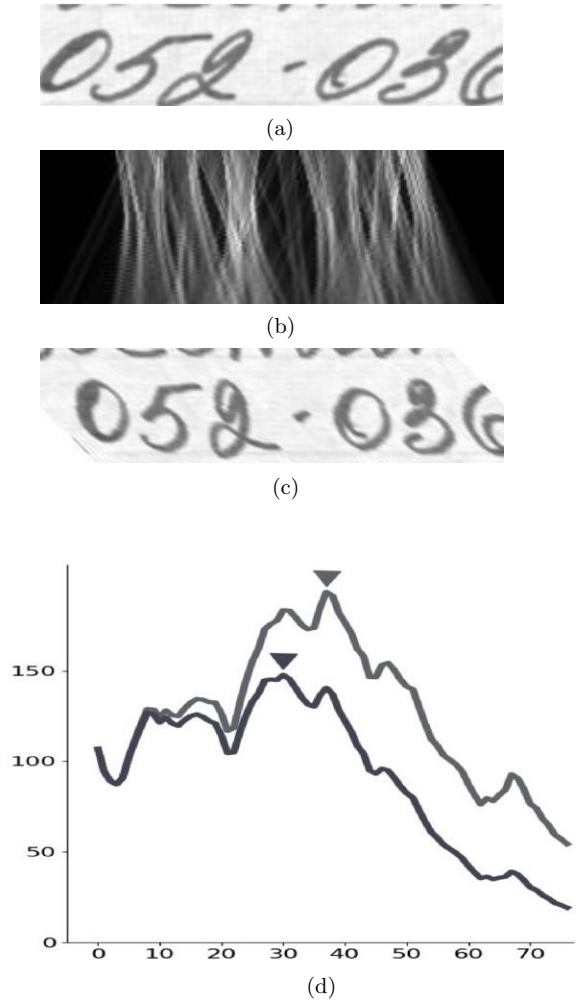


Figure 24.19. Textual block samples: (a) source image; (b) HT from horizontal derivative of (a); (c) result of rectification for (a); (d) $SSG_{FHT}(t)$ (darker) and $SSG_{DRT}(t)$ (lighter) plots for picture (b).

24.6.3 *Synthetic data generation*

In this section, we consider training data creation since it is widely used in neural networks' training for both symbols' recognition and segmentation.

Algorithm 1 Textual block slant correction algorithm

procedure TBSC(I, R)
    Input: Input grayscale image I
    Output: Corrected grayscale image R
    Downscale source image I twice on horizontal axis;
    Calculate horizontal derivative for downscaled bitmap and store it as image D;
    Calculate $FHT_{\text{image}}(D)$;
    for all $i \in rows(FHT_{\text{image}}(D))$ **do**
        Calculate $SSG_{FHT}(i)$;
        Calculate $k_i = \sqrt{1 + i^2/(h-1)^2}$
        Approximate $SSG_{DRT}(i) \approx k_i^3 \cdot SSG_{FHT}(i)$;
    Select row index i_{res} with maximal criterion value SSG_{FHT};
    Convert row index i_{res} into slant t_{res};
    Restore original slant $t'_{\text{res}} = 2 \cdot t_{\text{res}}$;
    Convert slant t'_{res} into angle α;
    Perform deslanting of an image I with angle $-\alpha$ into image R

Modern methods of synthetic data creation can be divided into three large groups: augmentation of real data, creation of synthetic data based on real data features, and creation of synthetic training datasets. Natural data augmentation consists of applying different distortions to dataset elements. The choice of distortions depends on the problem being solved. Successful usage of data augmentation is described in [97]. In these articles, the authors augment natural datasets with such distortions as homography, blur, intensity change, etc. The authors in [60,65] also use natural data augmentation in the task of machine-readable zone (MRZ) recognition. Yet while being a sufficient method, dataset augmentation requires natural data. For instance, in [97], the best quality of MRZ symbols' recognition is achieved with 20 natural examples of each symbol, received from scanned images. Additionally, overuse of augmented images can lead to the loss of recognition quality.

In this chapter, we consider the task of data generation in the context of Internal Russian passport fields' recognition. This type of document has several traits that should be noted, which are important for the training datasets. First, unlike MRZ mentioned in [65, 97], there is no standard font for Russian passports. Also, paper quality and printing devices vary depending on the region and year of issue.



Figure 24.20. Examples of letters "A" from Russian national passport.

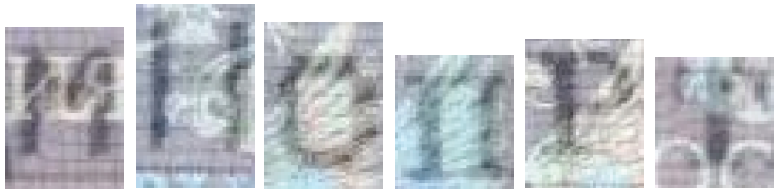


Figure 24.21. Examples of symbols with glares and parts of holographic elements.

Consequently, Russian national passports differ greatly (Figure 24.20). Besides, passport has different protection measures, such as laminating film, on which holographic elements and glares appear (Figure 24.21). In addition, personal information from Russian national passports is protected by law, forbidding its gathering and usage. In these circumstances, gathering natural dataset with all possible variants becomes a really complicated and even impossible task.

The general data generation scheme is described in more detail in [39]. The flow chart on Figure 24.22 presents the general scheme of the algorithm.

We evaluated our method for synthetic data generation using a test dataset of natural images consisting of 18 symbols with 1500 images per symbol. Each sample of test dataset was a separate image of letter which allowed us to measure the recognition quality without accounting for the errors of other subsystems (e.g. segmentation). As a baseline, we trained convolutional neural networks, using natural training dataset with 10467 images per symbol obtained from images of real Internal Russian passports. The recognition error of natural CNN was 0.2%. We considered 18 symbols of Cyrillic alphabet most frequently met in Russian passports because the number of examples of the other 15 letters is not enough for training and test datasets.

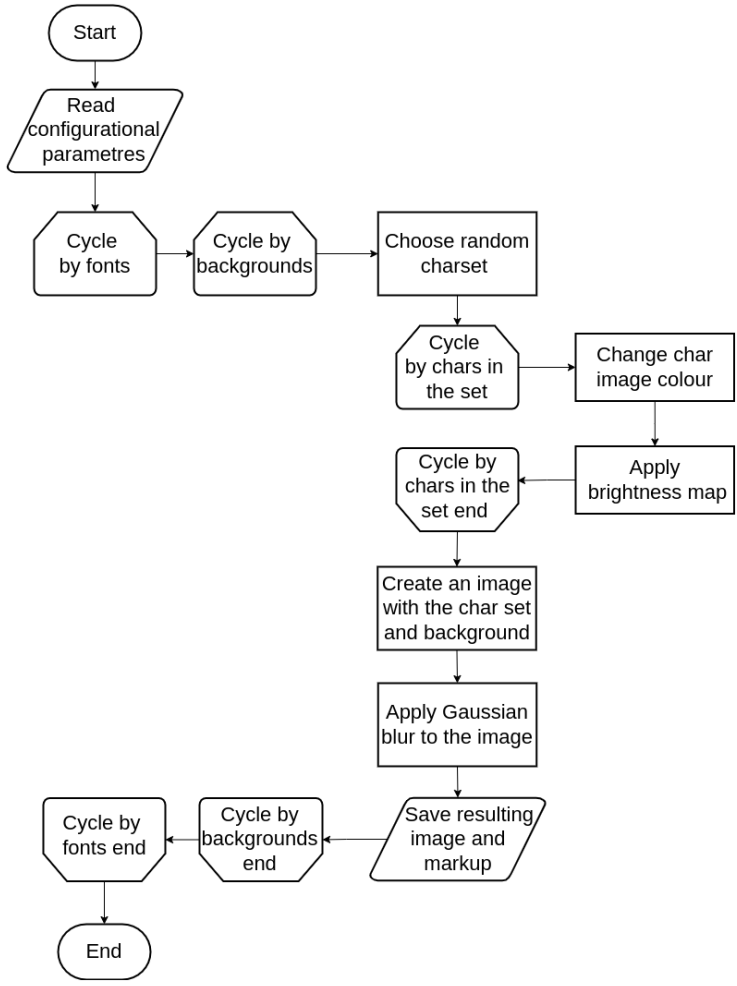


Figure 24.22. Main steps of the algorithm for string image synthesis.

We used 40 fonts and 37 backgrounds as an initial generation variety. We created 10 training datasets with different distortions: 5 without background and 5 with it. For each group of datasets, we gradually added distortions. The first dataset contained images without distortions, and then we applied Gaussian blur. After that, we added vertical and horizontal shifts and projective transformations. Brightness map was the last added distortion. We tested our networks on the natural test dataset. The result is demonstrated in Table 24.9. The best error for created datasets — 3.85% — was achieved on the

Table 24.9. CNN training results on initial generation variety. Each of the following datasets contains all previous distortions.

Dataset	Data without background		Data with background	
	Train error (%)	Test error (%)	Train error (%)	Test error (%)
Binary letter images	0	23.46	0.01	13.55
+Gaussian blur	0	26.51	0.02	12.74
+Shifts	0	45.24	0.09	5.06
+Projective transformation	0	22.63	0.09	3.98
+Brightness map	0.06	26.38	0.09	3.85
First stage of fonts addition	0.03	17.16	0.09	2.35
Second stage of fonts addition	0.02	13.76	0.09	1.2
Natural training data	—	—	0.08	0.2



Figure 24.23. Example of segmentation and recognition of natural string in German.

one with backgrounds and all distortions. CNNs trained on datasets without backgrounds demonstrated zero error on the train set and high variance on the natural test, which is connected with low relevance of the training data to the natural test and a high overfitting due to low variability of the training data.

The achieved error rates are displayed in Table 24.9. Error for synthetic datasets with background is 2.35% after the first stage of addition and 1.20% after the second stage. The final quantity of fonts is 111. After achieving sufficient results on Russian data, we tested our system on real ID documents from Germany and Georgia, to demonstrate applicability of our method for different alphabets. Our choice of those two languages stemmed from the features of their alphabets. The German alphabet consists of Latin letters and contains symbols with diacritical marks, which complicates its recognition. Figure 24.23 demonstrates successful use of suggested method for alphabets with letters with diacritics. Yet the Cyrillic and Latin symbols are rather similar, so we also tested our system

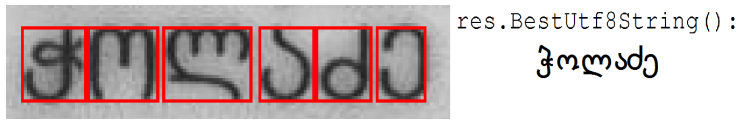


Figure 24.24. Example of segmentation and recognition of natural string in Georgian.



Figure 24.25. Examples of images for recognition.

on the Georgian language. It has its own alphabet, which script, known as Mkhedruli, differs from the Latin and Cyrillic ones and corresponds to the Greek alphabet. Figure 24.24 demonstrates the segmentation and recognition results on a line from the Georgian ID card.

The described method is also applicable for more complicated cases when the alphabet size is much greater (for example, Hangul [66]). But in this case, the embedded networks must be used instead of classification neural networks.

24.6.4 *Training data augmentation*

To train neural networks, it is very important that the training dataset is balanced and has the same data distribution as the data we are going to recognize [140]. There are some examples of images from symbol recognition problems in Figure 24.25. The first column consists of ideal images and the others show examples from the real word, which are sometimes distorted and blurred. The neural network will have high probability of failure on distorted images if we train it using only ideal images.

There is no strong line between static data augmentation before training and "on-line" augmentation in the training process. In this section, we speak about "on-line" augmentation since it has unarguable advantages in comparison to static recognition: with the same initial training dataset, the neural network can be trained using a vast variety of augmented images without spending a lot of space to store them.

If X^{train} is a training dataset consisting of N images of the same size and Y^{train} is a set of respective labels,

$$X^{\text{train}} = x_1, x_2, \ldots, x_n. \tag{24.18}$$

Let us consider a training process of the model M in separated moments. Φ is a vector of parameters and L is a loss function. Quality of the model is evaluated using the test dataset X^{test} and its labels Y^{test}. The best achieved quality on the test dataset can be described with equation (24.19):

$$\begin{aligned} Q(M, X^{\text{train}}, Y^{\text{train}}) &= \min_{r_{t_i}}(M_{t_i}(X^{\text{train}}, Y^{\text{train}}, X^{\text{test}}, Y^{\text{test}}, \Phi_{t_i}, L) \\ &= r_{t_i}), r_{t_i} \in \mathbb{R}; i = \{1, 2, \ldots, K\}. \end{aligned} \tag{24.19}$$

Our task is to specify a set of transformations $F = \{f_1, f_2, \ldots, f_D\}$, where

$$\begin{aligned} f_i = f_i(x_j) = \breve{x_j}, \\ Y(x_j) = Y(\breve{x_j}). \end{aligned} \tag{24.20}$$

Random subset $F' \subseteq F$ is be applied to X^{train} (we refer to the transformed train dataset as $F'_{t_i}[X^{\text{train}}]$):

$$\begin{aligned} Q(M, F'_{t_i}[X^{\text{train}}], Y^{\text{train}}) < Q(M, X^{\text{train}}, Y^{\text{train}}), \\ i = \{1, 2, \ldots, K\}. \end{aligned} \tag{24.21}$$

Every function f_j has a probability $p(f_i \in F'_{t_i}) = \frac{1}{p}$ to be chosen. Subset of chosen functions is F'_{t_i}.

Final algorithm takes a set of images as an input and transforms it according to the list of possible transformations from the parameters. Every image is transformed with a random subsample of transformations in random order. In this algorithm, we used the most common transformations for the symbol recognition problem: projective transformations, image cropping (with consequent scaling to the initial

size), horizontal and vertical flips (in some specific cases), Gaussian blurring, addition of lines, cyclic shift, small rotation, addition of noise, brightness modifications [48], and morphological operations.

Since we consider on-line augmentation, it is very important to make it efficient and maintain acceptable training time. This is achieved using a parallel scheme: while the GPU trains neural network on the set of the images, the next set is being prepared using the CPU. In this case, we can fully utilize the CPU which is typically idle in the training process. Using this trick we have managed to maintain training time unchanged since the data augmentation was not slower than training. The method effectiveness in terms of trained neural network quality was tested using MNIST dataset and neural network architecture SimpleNet(310K). The resulting error rate decreased from 0.28% to 0.25%. Close result was achieved with 4 times lighter neural network architecture: error rate decreased from 0.28% to 0.26%.

24.6.5 *Symbols' segmentation*

String to symbols' segmentation is a very important stage of the optical character recognition (OCR) task. Segmentation algorithms take the image of a string as an input and return positions of the cuts between symbols or symbol bounding boxes.

Importance of the segmentation is assured with the fact that most recognition algorithms take separate symbols instead of strings or words. In very rare cases (when recognizing documents with known symbols' positions), symbols' positions can be defined, but in general case, we have to find them in runtime. Sometimes, this task can be simplified if we deal with monospace fonts. In general, a big part of the recognition errors comes from incorrect segmentation [31].

In this section, we consider different methods for text fields' segmentation using a system of Internal Russian passports' recognition as a framework. Since this system can work in real-time on mobile devices, segmentation algorithm should be fast enough and therefore it is not possible to use algorithms with high computational complexity. Another requirement is that segmentation method has to be robust against different distortions and lighting conditions (Figure 24.26). Segmentation quality estimation is a separate task which can be solved in many ways (e.g. root mean square distance between the found and ideal cuts, precision, and sensitivity). In this



Figure 24.26. Examples of strings from the Internal Russian passport recognition system.

section, we evaluate this quality as the recognition quality which system yields with the considered segmentation algorithm since this is the most important property of the system. The main drawback of this method is that the ideal quality of the segmentation method is limited with other subsystems' imperfections, but still we are able to compare algorithms with each other.

Engineering approach: Segmentation process starts with naive cuts placing: the image is projected to a horizontal axis so that at each point there is a minimum brightness value for the corresponding column of the picture. Then, the recursive procedure of the projection cutting is launched. In a perfect case, every symbol corresponds to one segment, which is defined with a position on the projection, width, and confidences of the corresponding cuts. The cutting procedure continues while there are segments wider than a certain predefined size. Typically, after this stage, incorrect cuts are presented: some of them are missed due to merged letter on the input image, while some of them are placed within the letter. The next stage converts cuts to symbol boxes and adjusts their borders to match symbols in the input image better. After that we run the most important step of the algorithm which is based on the principles of dynamic programming. It tries to find the optimal symbols' positions according to the cuts' confidences and the results of special neural network, which was trained with special symbol "incorrectly segmented letter". Step-by-step results of this algorithm are shown in Figure 24.27.

Convolutional NN: The used neural networks take gray value images as input feature vectors, gathered from images of Russian passports. These images were scaled to 200×20 before passing to

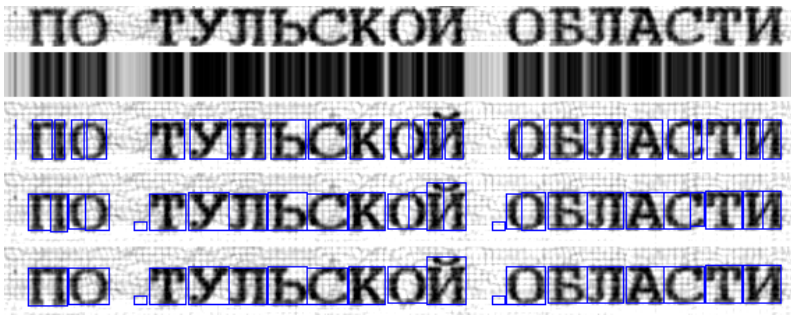


Figure 24.27. Steps of text segmentation: filtered image, projection with minimum, naive cuts, symbols boxes, and merged boxes.

the neural network. The size of the final vector is also 200 and every value corresponds to the cut confidence at a given position. The neural network consists of convolutional and subsample layers followed by two fully connected layers. Activation function was hyperbolic tangent.

Test of this neural network showed promising results (see Table 24.10), but it was unable to beat engineering approach described in the previous paragraph. To further improve accuracy, we used the result of this neural network as an input to an additional recurrent neural network.

Recurrent NN: The used bi-directional LSTM neural networks take as an input a sequence of n previous results from the convolutional neural network, therefore at every input position, there are n neighboring answers. Bi-directional neural network actually consists of two single-directional neural networks. One of them processes image in left to right direction, another one in right to left. Outputs of both single-directional neural networks are concatenated and then passed to the fully connected layers. It is important to note that the convolutional part of this network system does not change during the training of the recurrent part. In our experiments, $n = 10$.

Figure 24.28 shows the intermediate and final results of this approach. Red background marks probabilistic values from the convolutional neural network, yellow shows the output of the recurrent neural network, and blue corresponds to the local maxima and represents the final result.

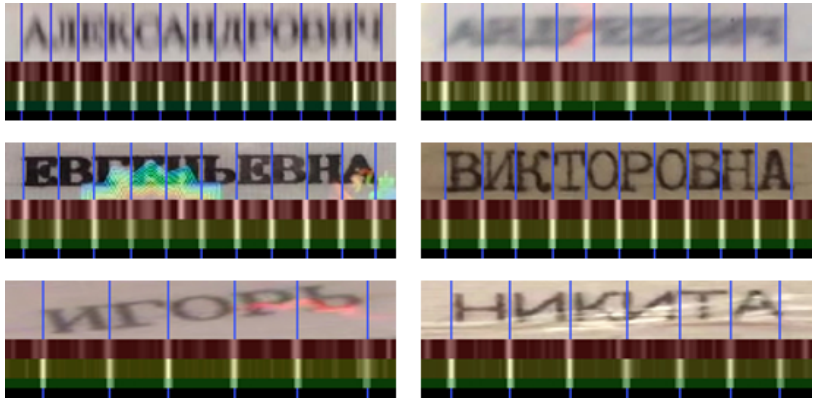


Figure 24.28. Segmentation result output of the Recurrent NN.

Fully convolutional NN: Unlike in the previous paragraph, it is possible to use fully convolutional neural networks in the task of segmentation. In this case, we have to scale an image to a constant predefined height, but the width of the image remains unchanged. Next step is to build a deep fully convolutional neural network architecture. There are several limitations to the architecture: the neural network has to contain layers without image size requirements (convolutional, pooling, and activation functions), the input image of $w \times h$ size should be collapsed to $w \times 1 \times 1$ size where every value represents confidence of the cut in a given position of the input image, and the number of trainable weights in the neural network should be as low as possible to reduce the possibility of overfitting and increase performance. The resulting neural network consisted of only 1741 weights which is very low. Another feature of the neural network in this experiment was the fact that it was trained on the synthetic data generated using a method described in [39].

To train this neural network, we minimized $L2$ norm between the output vector and the ideal vector. Results were not good enough, but their analysis showed two main reasons for it. The first reason was data imbalance. There were much more positions without a cut than with it, therefore the trained neural network tended to miss cuts since it was better in terms of the loss function. The second problem was that the loss function did not take into account the distance between ideal and resulting cuts. Even the error of one pixel was the



Figure 24.29. Ideal and real vectors (left) and their cumulative summations (right).

Table 24.10. Comparison of the segmentation methods.

	Per-string quality (%)
Engineering approach	88.7
Convolutional NN	74.2
Recurrent NN	89.3
Fully convolutional NN	86.3
Fully convolutional NN with CS	91.0

same as the error of ten pixels, for example. This behavior is illogical and does not represent our expectation from the neural network.

Fully convolutional NN with CS: To solve the problems described in the previous paragraph, we developed a cumulative sum layer. In this case, we minimized $L2$ norm not between the ideal and the received answers but between their cumulative sums. With this modification, we introduced two specific ideas to neural network learning criteria: the "number" of placed cuts must be approximately the same and small deviations in cuts' positions are much less important than big ones. Figure 24.29 shows four vectors. In classical $L2$ norm optimization, we minimize the distance between the vectors from the first row, while after cumulative sum layer application, we should minimize the distance between the vectors from the second row.

Segmentation methods' comparison: Comparison of the described segmentation methods is presented in Table 24.10. It can be seen that fully convolutional neural network with cumulative sum is the best one in terms of the recognition quality. It also can be applied to strings of any sizes since it does not include scaling to

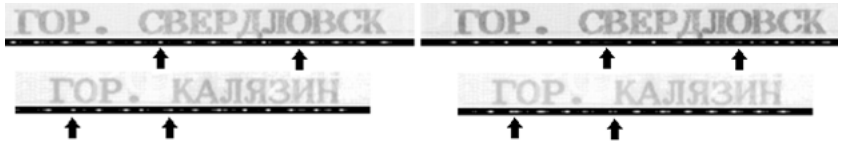


Figure 24.30. Comparison between fully convolutional neural networks with and without cumulative sum.

constant width step (as convolutional and recurrent neural networks do). Figure 24.30 shows comparison between fully convolutional neural networks with and without cumulative sum.

Using SIMD architectures: Another way to improve performance on modern CPUs is to use SIMD extensions. SIMD (Single Instruction, Multiple Data) is a class of parallel hardware. Such computers perform the same operation on the elements of data vector simultaneously. Although there are SIMD extensions for various CPUs, algorithms are often not vectorized automatically. This is due to the fact that compiler optimization is not able to change program behavior and data types used. As a result, many algorithms have huge vectorizing potential. In this chapter, we show how to improve neural networks' performance with the help of SIMD vectorization and fixed-point arithmetic. The research was done on Samsung Exynos 5422, ARM NEON.

IEEE 754–2008 standard specifies compact half-precision floating-point format that occupies 16 bits. It is used for data which do not require high accuracy. It consists of 16 bits: sign bit, 5-bit exponent with a bias of 15, and 10-bit mantissa. Calculations in half float type are not supported by modern CPUs, so half float is converted to float before calculations and then back. There is a fast conversion from float to half float and reverse algorithm described in [134]. A number of modern ARM CPUs support half float data type and fast conversions to float and backward. We made an assumption that conversion takes less time than data loading from memory and usage of half float data type speeds up calculations. Since Eigen library does not support half float, we added it as a custom data type and vectorized half float operations. It resulted in matrix addition and coefficient-wise multiplication speed-up, although matrix multiplication slowed down. Recognition also slowed down, which means our assumption

Table 24.11. Comparison of matrix operations' time in half float, fixed, and float data types on Samsung Exynos 5422.

	Matrices sizes			
	10×10	128×128	512×512	1024×1024
	Additional			
half float	4.44 μs	567.00 μs	9.25 ms	45.90 ms
fixed	3.56 μs	369.00 μs	6.10 ms	32.00 ms
float	5.36 μs	775.00 μs	15.30 ms	67.30 ms
	Coefficient-wise multiplication			
half float	5.37 μs	609.00 μs	10.00 ms	47.70 ms
fixed	4.39 μs	426.00 μs	7.00 ms	35.70 ms
float	6.34 μs	834.00 μs	16.30 ms	72.70 ms
	Matrix multiplication			
half float	61.00 μs	62.70 ms	3.93 s	31.10 s
fixed	32.10 μs	22.80 ms	1.36 s	10.80 ms
float	34.40 μs	39.50 ms	32.51 s	10.80 s

was wrong (see Table 24.11). It appeared that regularized fully connected ANNs do not lose accuracy with half float data type. It means that such calculation precision decrease does not influence the result.

An alternative way is to use integer SIMD arithmetic [129, 130, 132], but it requires new methods for ANN training [112]. The next approach to increasing the speed of neural networks is abandoning multiplication. But detailed description of this technique is beyond the scope of this chapter and can be found in [83, 84].

24.7 Integration and Stopping

24.7.1 *Results' integration*

Modern mobile devices are equipped with high-quality cameras and decent computing power which allows them to be used for camera-based document analysis tasks. Document analysis on mobile devices presents a set of challenges, such as motion blur, defocus, glares on reflective document surfaces, insufficient camera resolution for accurate OCR, and others [10, 35, 37]. However, the possibility to

process multiple frames in real time can be used to mitigate some of these difficulties. In particular, having the ability to recognize the same object multiple times in different video frames can significantly increase the overall recognition accuracy. Multiple recognition results for the same document object (e.g. text field) can be either sorted in some order of preference or confidence and then a single result with the highest expected accuracy can be selected, or the results can be integrated together to produce the result which could be better than each individual result.

From the system's composition perspective, the process of video stream OCR which independently processes each frame and integrates the recognition result can be regarded as an anytime algorithm [148]. From the desired properties of anytime algorithms, one can expect *interruptibility* (i.e. the process may be stopped after any frame and currently integrated result will be available as the answer) and *monotonicity* (i.e. on average the quality of integrated results does not decrease). However, the recognition result, at least in the case studied in this section, does not have a *recognizable quality*, i.e. the quality of the current result cannot be determined at run time. It is worth mentioning that for OCR problems, there exist other classes of anytime algorithms, such as algorithms which output partial recognition results: first, for characters which are easy to recognize, and progressing to the most difficult characters, recognition of which demands higher computation time [73]. Although different from the task considered in this section, for such cases, the problem of optimal stopping is also relevant and also requires attention.

Thus, the integration between frames is one of the most important parts of the document recognition in video stream. Consider, for example, the task of text field recognition: Figure 24.31 demonstrates an example of how text field images may appear on consequent frames of a mobile document video stream. Let $\mathbb{X}$ be the set of all possible text field recognition results, with a defined normalized metric function $\rho : \mathbb{X} \times \mathbb{X} \to [0, 1]$. For each document in a document recognition session and for each of the document's field with a correct value X^*, a sequence of random recognition results $\mathbf{X} = (X_1, X_2, \ldots)$ is observed one result at a time, each observation $x_i \in \mathbb{X}$ is a realization of X_i. Let's now define recognition results' integrator as a function of several recognition results which produces

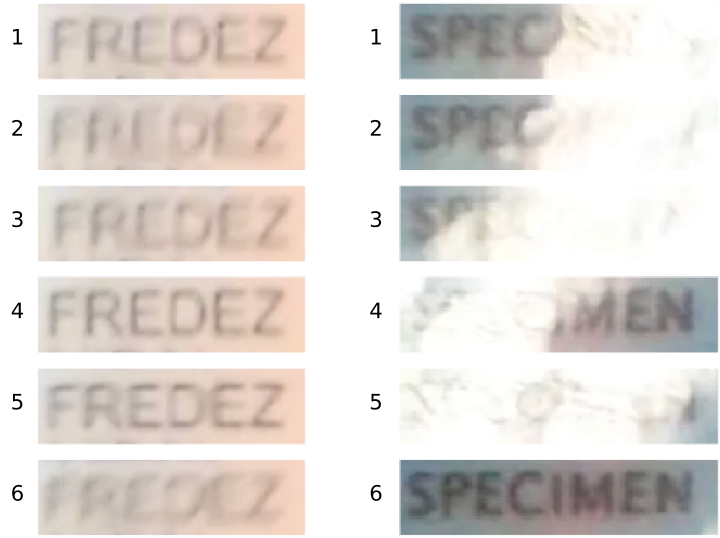


Figure 24.31. Examples of text field frames as appeared in the video stream. Images were taken from MIDV-500 dataset [6] (clip TS 07 field 1 and clip HA10 field 2).

a single integrated result $R : \mathbb{X}^+ \to \mathbb{X}$ (here by $\mathbb{X}^+$ we mean the set of all non-empty sequences of elements from $\mathbb{X}$). At any moment n, the observations $X_1 = x_1, \ldots, X_n = x_n$ are obtained and the integrated result $R_n \stackrel{\text{def}}{=} R(x_1, \ldots, x_n)$ can be produced.

The problem of recognition results' integration is now to find a function R such that it minimizes the expected distance between the integrated result R_n and the correct result X^*:

$$R^* = \arg\min_R \mathrm{E}\left(\rho(R_n, X^*)\right). \tag{24.22}$$

The metric ρ should be selected such that it represents the similarity between the correct value of the recognized object and its recognition result. The most basic metric function can be defined as a strict equality. The average value of this metric on a reference dataset is in inverse proportion of the recognition accuracy:

$$\rho_E(x, y) \stackrel{\text{def}}{=} \begin{cases} 0 & \text{if} \quad x = y, \\ 1 & \text{if} \quad x \neq y. \end{cases} \tag{24.23}$$

For text field recognition, an alternative, more detailed metric function can be defined as a Normalized Levenshtein Distance

ρ_L [144]:

$$\rho_L(x,y) \stackrel{\text{def}}{=} \frac{2 \cdot \text{levenshtein}(x,y)}{|x| + |y| + \text{levenshtein}(x,y)}, \tag{24.24}$$

If there are no character recognition confidence estimations available (i.e. if $\mathbb{X}$ is a set of strings over some fixed alphabet $\mathbb{A}$), the most universal approach to the recognition results' integration is the ROVER (Recognizer Output Voting Error Reduction) method [57], which is used for merging text field recognition results produced with different recognition algorithms [124, 141] and for accumulating text recognition result in video stream [27,102]. The algorithm consists of two modules: the alignment module performs an optimal wrapping of each incoming string to a word transition network, then the voting module selects the best result by traversing the network and choosing the best result on each stage. In case of strings' integration, the transition network can be built by a Levenshtein-based optimal wrapping of the incoming string, and the voting can be frequency-based. To implement such a method, an empty symbol should be defined with a fixed voting weight relative to the weight of each symbol of the alphabet $\mathbb{A}$. The ROVER algorithm could also be extended to incorporate per-character membership estimations [18] and weighting of individual frames [103].

Table 24.12 lists the number of unique fields in MIDV-500 dataset, the total number of field images (across all frames with fully visible document), and average field image sequence size, for the four major field groups (numeric dates, document number, MRZ lines, and Latin name components). Only the frames on which the document is fully visible were considered, thus the clips in the analyzed dataset had different lengths (from 1 to 30 frames). In order to minimize the extra normalization effects and provide a more clear presentation of the results, each evaluated clip was extended up to the length of 30 using repetitions from the beginning of the clip (thus, all evaluated clips for the performed experiments had the same length). The table presents average distances from the frame result X_i to the correct result X^*, and from the integrated result for the whole clip to the correct result X^* before extension (R_{last}) and after extension (R_{30}), for distance metrics ρ_E and ρ_L.

Each field was cropped according to the combined ground truth of document boundaries and template text field coordinates, with

Table 24.12. Average text field recognition result metrics for Tesseract [123] on MIDV-500 dataset [6]. X_i is the result from any frame, R_{last} is the full clip result produced using ROVER [57], and R_{30} is the extended clip result produced using ROVER [57].

	Numeric dates	Doc. numbers	MRZ lines	Names (Latin)	All fields
Unique fields	91	48	30	79	248
Clips in total	824	436	260	719	2239
Images in total	17735	9329	5096	15587	47747
Aver. clip size	21.523	21.397	19.600	21.679	21.325
Tesseract v3.05.01:					
$\mathrm{E}(\rho_E(X_i, X^*))$	0.792	0.791	0.912	0.759	0.794
$\mathrm{E}(\rho_E(R_{\mathrm{last}}, X^*))$	0.660	0.702	0.842	0.629	0.679
$\mathrm{E}(\rho_E(R_{30}, X^*))$	0.664	0.697	0.850	0.629	0.681
$\mathrm{E}(\rho_L(X_i, X^*))$	0.360	0.422	0.258	0.443	0.388
$\mathrm{E}(\rho_L(R_{\mathrm{last}}, X^*))$	0.244	0.326	0.162	0.338	0.281
$\mathrm{E}(\rho_L(R_{30}, X^*))$	0.246	0.323	0.164	0.336	0.280
Tesseract v4.0.0:					
$\mathrm{E}(\rho_E(X_i, X^*))$	0.570	0.586	0.940	0.475	0.581
$\mathrm{E}(\rho_E(R_{\mathrm{last}}, X^*))$	0.424	0.440	0.904	0.293	0.441
$\mathrm{E}(\rho_E(R_{30}, X^*))$	0.426	0.447	0.904	0.295	0.444
$\mathrm{E}(\rho_L(X_i, X^*))$	0.238	0.287	0.339	0.250	0.262
$\mathrm{E}(\rho_L(R_{\mathrm{last}}, X^*))$	0.123	0.160	0.277	0.125	0.149
$\mathrm{E}(\rho_L(R_{30}, X^*))$	0.125	0.163	0.279	0.127	0.151

added margins equal to 10% of the smallest text field dimension. Each text field cropping size corresponded to 300 DPI resolution and was recognized using open-source text recognition engine Tesseract [123] (versions v3.05.01 and v4.0.0) using default parameters for text line recognition in the English language. All character comparison was case-insensitive and the Latin letter *O* was treated as equal to the digit 0. In the performed experiments, we assumed the weight of empty symbol to be 0.6 in the ROVER implementation. Figure 24.32 illustrates the average distances from the integrated results to the correct result fields recognized by Tesseract OCR engine.

If the recognition results' confidences are somehow available (i.e. if each field recognition result is represented as a sequence

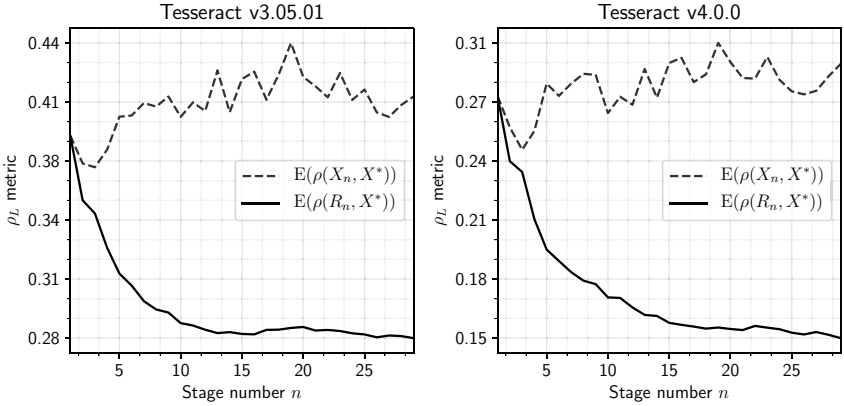


Figure 24.32. Average distances from the frame results and from the integrated results to the correct values, for distance metric ρ_L, as measured on text fields taken from MIDV-500 dataset [6].

of m character results $x_i = \{A_1^i, \ldots, A_m^i\}$ where each character result A_j^i is a mapping $A_j^i : \mathbb{A} \to [0, 1]$ from alphabet $\mathbb{A}$ to their pseudo-probabilistic estimations, with $\sum_{s \in \mathbb{A}} A_j^i(s) = 1$), the another approach to recognition results' integration is the selection of the "best" result from a sequence of given hypotheses, based on a predefined criterion — "reliability predictor". Simple yet a well-performing predictor is an average difference between the two highest alternatives:

$$\pi_{\text{avgdiff}}(x_i) = \frac{1}{m}\sum_{q=1}^{m} \max_{s \in \mathbb{A}} A_q^i(s) - \frac{1}{m}\sum_{q=1}^{m} \max_{s \in \mathbb{A} \setminus \arg\max\limits_{g \in \mathbb{A}} A_q^i(g)} A_q^i(s). \tag{24.25}$$

More robust predictors may be constructed for specific character recognition CNNs as linear functions of low-level predictors based on alternative values [7] or using language models for specific text fields. The method of selecting a single frame recognition result is not suitable in cases when each frame yields only partially recognized fields (e.g. for documents with holographic layer which obstructs parts of the text field on several frames). In this case, a combination of field recognition results is performed using averaging of A_q^i mappings (e.g. within the voting module of the ROVER algorithm) using reliability predictors $\pi(x_i)$ and character image quality estimations as weights.

24.7.2 *Stopping rules*

Although there are some published works mentioning multiple OCR results integration [22,27] and the video stream OCR stopping problem [5, 19], the corpus of tested and evaluated methods related to this task is hardly sufficient. At the same time, there exists a large amount of works dedicated to more general stopping rule problems in mathematical statistics and decision theory [13, 40, 53, 127]. This includes such well-researched problems as the secretary problem [93], house-selling problem [55], S_n/n^1 problem or the problem of maximizing the average [71], and others. The proofreading problem [53,54] can be regarded as particularly relevant to the text field recognition in video stream. This problem goes as follows: a manuscript has been digitized with some number of errors M. A proofreading can be performed for the digitized version, and each i-th proofreading corrects X_i mistakes and costs a fixed amount c. Each mistake present in the final version of the text also has an inherent cost. The task is to make a decision after i-th proofreading, whether the current version of the digitized manuscript should be published or the review process should continue, in order to minimize the total expected cost. A number of solutions is proposed for this problem [54, 143], based on different assumptions about the distribution of M and X_i. This problem has several variations and other applications, e.g. to software testing [43].

A certain similarity can be observed between proofreading stopping rule problem and the problem of stopping the video stream recognition process: the text field is recognized on multiple frames such that to obtain the recognition result for next frame some cost must be paid (which could represent the time required to pre-process the next frame and perform another text recognition iteration). Assuming that each time a single accumulated recognition result is defined, after each frame, a decision must be made either to pay the additional observation cost and continue the recognition process in hope that the result would improve, or to stop the process and output the currently accumulated result. In this case, the expected loss can be represented as a linear combination of expected number of processed frames and a distance between expected recognition result to the correct text field value (in terms of some pre-defined metric).

There are important differences between a video stream recognition stopping rule problem and the proofreading problem, which have to be considered as follows:

(1) In most formulations of the proofreading problem, the value of X_i is taken to be non-negative, i.e. it is assumed that each proofreading either corrects some mistakes or at least doesn't introduce any new ones. In the text field recognition problem, there is no guarantee that the recognition result on the next frame, or the integrated result after processing the next frame, will always be closer to the correct value. However, it can be assumed that the integration algorithms are usually designed in such a way that the recognition of multiple versions of the object generally has higher accuracy than the recognition of a single image of an object.
(2) The solutions for the proofreading problem or its variations typically rely on the decision-maker having the ability to assess either the loss which will be suffered if the stopping decision is made or the difference in losses between stages, as the observed value of X_i directly contributes to the loss function. In the case of text recognition in video stream, however, the distance from the currently obtained result to the correct result could only be assessed using some recognition confidence estimations (which could suffer from overconfidence [28, 101]) or could even be completely unavailable.

As in the previous section, consider that the sequence of observations of some random recognition results $\mathbf{X} = (X_1, X_2, \ldots)$ is observed one result at a time, and the process may be stopped at any time $n > 0$ with a cost:

$$L_n \stackrel{\text{def}}{=} \rho(R_n, X^*) + c \cdot n, \tag{24.26}$$

where R_n is the integrated result produced given the first n observations, using the integration function (24.22), and c is some pre-defined positive observation cost. The loss associated with stopping at time $n = 0$ (i.e. not taking any observations at all) may be regarded as infinite. The task is to choose a time to stop the observations in order to minimize the expected loss. Let us formalize the problem using the notation used in [53]. A stopping rule can be defined as a sequence

of functions:

$$\Phi \stackrel{\text{def}}{=} (\phi_0, \phi_1(x_1), \phi_2(x_1, x_2), \phi_3(x_1, x_2, x_3), \ldots), \tag{24.27}$$

where $\forall n : 0 \leq \phi_n(x_1, \ldots, x_n) \leq 1$. The function $\phi_n(x_1, \ldots, x_n)$ represents a conditional probability of stopping at the stage n given that stage n have been reached (i.e. given that the values $X_1 = x_1, \ldots, X_n = x_n$ have been observed).

Based on the stopping rule Φ and the sequence of observations $\mathbf{X}$, a random stopping time N is determined as follows:

$$\begin{aligned}
&P(N = 0|\mathbf{X} = (x_1, x_2, \ldots)) = \phi_0,\\
&P(N = n|\mathbf{X} = (x_1, x_2, \ldots))\\
&\quad = \phi_n(x_1, \ldots, x_n) \times \prod_{j=1}^{n-1} (1 - \phi_j(x_1, \ldots, x_j)) \quad \forall n \in \{1, 2, \ldots\},\\
&P(N = \infty|\mathbf{X} = (x_1, x_2, \ldots)) = 1 - \sum_{j=0}^{\infty} P(N = j|\mathbf{X} = (x_1, x_2, \ldots)).
\end{aligned} \tag{24.28}$$

In terms of random stopping time N, the stopping rule for $n \in \{0, 1, \ldots\}$ is expressed as

$$\phi_n(X_1, \ldots, X_n) = P(N = n|N \geq n, \mathbf{X} = (x_1, x_2, \ldots)). \tag{24.29}$$

Both the sequence of functions Φ and the random stopping time N can be used interchangeably to represent the stopping rule. The problem of optimal stopping is to choose a stopping rule N to minimize the expected loss $V(N)$:

$$V(N) = \mathrm{E}(L_N(X_1, \ldots, X_N)). \tag{24.30}$$

Let's consider this stopping rule problem as a monotone one [41, 53]. Let $\mathrm{E}_n(\cdot)$ denote a conditional expectation $\mathrm{E}(\cdot|X_1 = x_2, \ldots, X_n = x_n)$. Let A_n denote the event $\{L_n \leq \mathrm{E}_n(L_{n+1})\}$. The stopping rule problem is said to be monotone if

$$A_0 \subset A_1 \subset A_2 \subset \cdots. \tag{24.31}$$

The condition (24.31) means that if at some stage n the loss function is not higher than the expected loss function at the next stage,

then this will be true for all future stages as well. Consider a stopping rule called one-stage look-ahead rule (also called a myopic rule):

$$N_A = \min\{n \geq 0 : L_n \leq \mathrm{E}_n(L_{n+1})\}. \tag{24.32}$$

Rule N_A calls for stopping at stage n if current loss is not higher than the loss which will be suffered if we stop at stage $n+1$. It can be shown [41, 53] that if the stopping rule problem is monotone and it has a finite horizon (i.e. for some fixed $T < \infty$, any stopping rule must stop when it reaches stage T), then the one-stage look-ahead rule (24.32) is optimal.

For solving the stopping rule problem (24.30) with the loss function (24.26), we propose to use one-stage look-ahead stopping rule under the assumption that the problem becomes monotone starting from a certain stage. Let's make the following requirement for the integrator function R: the expected distances between two consecutive integrated recognition results decrease over time:

$$\mathrm{E}(\rho(R_n, R_{n+1})) \geq \mathrm{E}(\rho(R_{n+1}, R_{n+2})) \quad \forall n > 0. \tag{24.33}$$

Let B_n denote the event $\{\mathrm{E}_n(\rho(R_n, R_{n+1}) \leq c)\}$ and consider the stopping rule problem starting from stage n on which B_n occurred for the first time. The events considered in condition (24.31) take the following form:

$$\begin{aligned} A_n &: \{\rho(R_n, X^*) + cn \leq \mathrm{E}_n(\rho(R_{n+1}, X^*)) + cn + c\} \\ &= \{\rho(R_n, X^*) - \mathrm{E}_n(\rho(R_{n+1}, X^*)) \leq c\}. \end{aligned} \tag{24.34}$$

Using triangle inequality, we can obtain

$$\begin{aligned} \rho(R_n, X^*) - \mathrm{E}_n(\rho(R_{n+1}, X^*)) &\leq \mathrm{E}_n(\rho(R_n, R_{n+1})) \\ \Rightarrow B_n \subset A_n \quad \forall n > 0, \end{aligned} \tag{24.35}$$

and using (24.33), we can assume $\forall n > 0 : B_n \subset B_{n+1}$. Hence, starting from stage n on which B_n occurred for the first time, events $A_n, A_{n+1}, A_{n+2}, \ldots$ hold as well, so the problem can be considered monotone starting from this stage.

Let's now consider stopping rule which tells the decision-maker to stop on the earliest stage on which B_n occurs:

$$N_B = \min\{n > 0 : \mathrm{E}_n(\rho(R_n, R_{n+1})) \leq c\}. \tag{24.36}$$

If N_B tells the decision-maker to stop at stage n, then N_A also stops at this stage, and if the problem becomes monotone starting

from stage n, the decision of N_A is optimal, so the optimal rule N^* also stops at this stage. Moreover, if $\rho(R_n, X^*) - \mathrm{E}_n(\rho(R_{n+1}, X^*)) > c$, rule N_B doesn't stop and neither does N^* following the principle of optimality [53]. Thus, assuming (24.33), N_B will not stop preliminary, and if it tells the decision-maker to stop, then it's optimal to stop.

As required by the stopping rule N_B, on n-th stage of the process, an estimation has to be provided for the expected distance between integrated results $\Delta_n \stackrel{\text{def}}{=} \mathrm{E}(\rho(R_n, R_{n+1}))$ given current observations $X_1 = x_1, \ldots, X_n = x_n$. We assume that the integrator function R can be used to compute such estimation. Such estimation can be obtained by modeling the integrated recognition result on the next stage assuming that the new observation will be close to the ones already obtained:

$$\hat{\Delta_n} \stackrel{\text{def}}{=} \frac{1}{n+1}\left(\delta + \sum_{i=1}^{n} \rho(R_n, R(x_1, x_2, \ldots, x_n, x_i))\right), \qquad (24.37)$$

where δ is an external parameter. In general, the selection of the method for estimating the next integrated recognition result (or the expected distance between it and the current result) might depend on the nature of the integrator function R and other specifics of the problem. Under some simplifying assumptions, the modeling (24.37) can be performed more efficiently, as described in [21].

Figure 24.33 illustrates the average decrease of $\mathrm{E}(\rho_L(R_n, X^*)) - \mathrm{E}(\rho_L(R_{n+1}, X^*))$ (the distance from the integrated recognition result to the correct result) over time, the expected distance Δ_n between consequent integrated results, and its estimation $\hat{\Delta}_n$. The figure shows that although the stopping rule N_B is a very rough approximation of N_A, there is a practical justification of assumption (24.33) and that $\hat{\Delta}_n$ (24.37) provides a decent real-time estimation of Δ_n starting from $n = 2$.

In order to evaluate the efficiency of stopping rules, a graphical plot can be constructed which would show the average stopping time and corresponding distance from the obtained result to the correct one as the observation cost c is varied. As a baseline, a simple counting stopping rule N_K was evaluated, which stops at a fixed stage K. Additionally, two variations of stopping rule explored in [5] were evaluated. Since the original paper relies on recognition

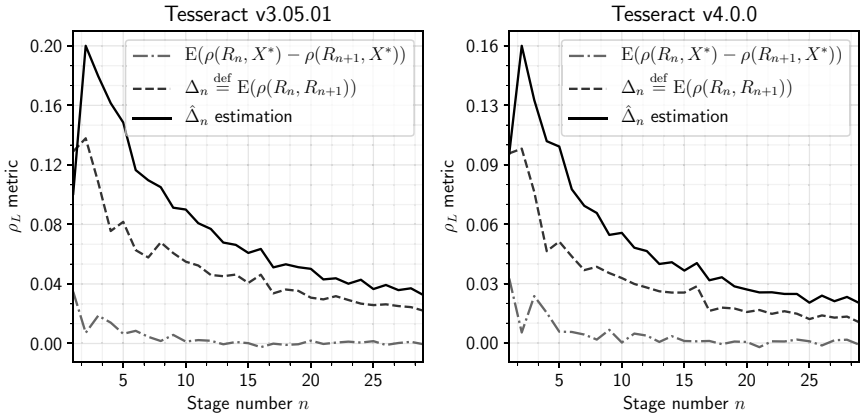


Figure 24.33. Average decrease of the distance between consequent integrated results and its proposed estimation, for distance metric ρ_L, as measured on text fields taken from MIDV-500 dataset [6].

result confidence estimations, which are unavailable in our experiments, the stopping rule described in [5] is reduced to thresholding at stage n the largest cluster of identical recognition results accumulated up to stage n. Thus, we constructed the stopping rule N_{CX}, which thresholds the largest cluster of identical frame recognition results among $x_1, \ldots, x_n$, and N_{CR}, which does the same for integrated results $R_1, \ldots, R_n$. Finally, the stopping rule N_B estimates at stage n the expected distance Δ_n to the next integrated result and stops when the estimation becomes lower or equal to a constant threshold. The stopping rule N_B only becomes active starting from stage $n = 2$ (i.e. when the estimation $\hat{\Delta}_n$ becomes more justified).

Figure 24.34 illustrates the stopping rules' efficiency for the four field groups when recognized in video stream using Tesseract recognition engine. The lower position of the curve signifies the better stopping rule performance. It can be observed that in general the proposed stopping rule N_B outperforms the other evaluated stopping rules in both metrics.

In the problem statement, it was assumed that each new observation has a constant cost c (24.26). In the real text field recognition system, this cost depends on the time required to produce the next recognition result, which may increase or decrease from stage to stage. Moreover, the observation cost may depend on the stopping

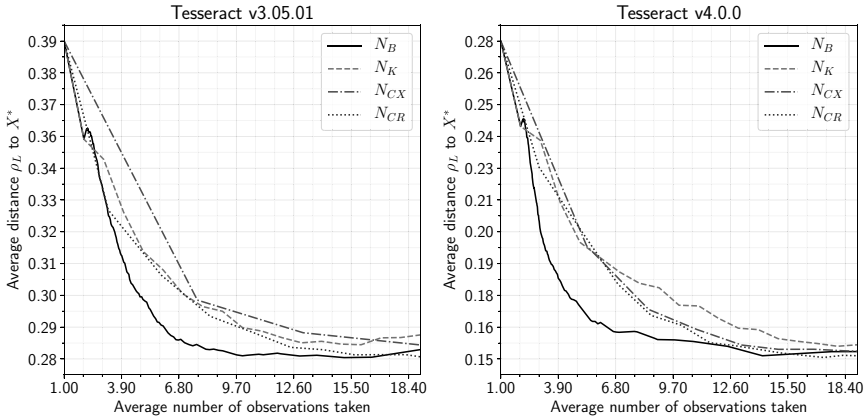


Figure 24.34. Stopping rules' efficiency evaluation: plot of the average distance from the obtained result to the correct one against the average number of processed frames as the stopping rule threshold is varied, for distance metric ρ_L, as measured on text fields taken from MIDV-500 dataset [6].

rule itself (e.g. include the time required to compute the estimation $\hat{\Delta}_n$) and on the stopping rules for other objects (e.g. the time required to obtain the next recognition result for a field may depend on whether the recognition process for other fields has stopped and thus on the next frame, those fields will not be recognized). It should be noted that assumption (24.33) is sufficient but not essential for obtaining $\forall n > 0 : B_n \subset B_{n+1}$, as it would also be sufficient to directly require that if starting from some stage n the expected difference $\mathrm{E}_n(\rho(R_n, R_{n+1}))$ becomes bounded by c, then it would be true for all later stages. Here we assumed (24.33), but a more general approach may be required if the model is to be generalized, such as classifier-based approaches [77]. For the sake of simplicity, the model doesn't include any kind of confidence estimations for the overall field recognition result or for separate characters (and thus it is assumed that the decision-maker doesn't have any grounds for estimating the distances $\rho(X_n, X^*)$ or $\rho(R_n, X^*)$. If the confidence estimations are available, the described model and the stopping rule N_B are still valid, however there could exist a better approximation of the one-stage look ahead rule N_A (24.32) which relies on text field recognition result confidence and on modeling the behavior of this confidence in video stream [8]. A more detailed evaluation of the

described stopping rules could be found in [24] and of their application for real-time document recognition systems in [20].

24.8 Conclusion

Methods described in this chapter make up together a framework for designing and implementing document recognition and analysis systems which operate on a mobile device. They have been successfully implemented in a commercial cross-platform document recognition system "Smart IDReader" [19]. Tables 24.13, 24.14, and 24.15 demonstrate recognition results of fields "Number", "Birth date", "Name", "Surname", and "Patronymic" for two distinct document types. Four measurements are presented as follows:

(1) *Per-frame recall*: Percentage of frames in the dataset on which the field was detected (i.e. not rejected by document matching, zone image quality evaluation, per-field segmentation, or field recognition subsystems).
(2) *Per-clip recall*: Percentage of video clips in the dataset for which the field was eventually detected (i.e. detected at least on one frame and not rejected by result integration, and final result analysis subsystems).
(3) *Per-frame accuracy*: Percentage of the frames in the dataset on which the field was detected *and* recognized without errors.
(4) *Per-clip accuracy*: Percentage of video clips in the dataset for which the field was eventually detected *and* integrated recognition result was correct.

In Table 24.13, field recognition results are presented for Russian driving licence, measured on a closed reference dataset containing 85 video clips with 1526 frames in total. Three distinct subtypes of Russian driving licence are present in this dataset. In Table 24.14, recognition results for the same set of fields are presented for Internal Russian passport, measured on a closed reference dataset containing 100 video clips with 3475 frames in total. All video clips for experiments in Tables 24.13 and 24.14 were captured using smartphones.

In Table 24.15, recognition results are presented for Internal Russian passport, measured on a closed reference dataset containing 96

Table 24.13. Fields' detection and recognition results for Russian driving licence (smartphone).

Doc. field	Per-frame recall	Per-clip recall	Per-frame acc.	Per-clip acc.
Number	62.06	82.35	93.67	98.58
Birth date	81.28	91.76	92.80	98.72
Name	89.32	94.12	90.88	95.00
Surname	89.72	96.47	87.12	98.78
Patronymic	88.57	93.42	82.40	90.14

Table 24.14. Fields' detection and recognition results for Internal Russian passport (smartphone).

Doc. field	Per-frame recall	Per-clip recall	Per-frame acc.	Per-clip acc.
Number	97.90	100.00	94.62	100.00
Birth date	91.94	100.00	95.16	96.00
Name	96.29	100.00	93.45	98.00
Surname	96.29	100.00	84.79	92.00
Patronymic	96.29	100.00	94.53	99.00

Table 24.15. Fields' detection and recognition results for Internal Russian passport (webcam).

Doc. field	Per-frame recall	Per-clip recall	Per-frame acc.	Per-clip acc.
Number	87.78	96.88	94.15	95.70
Birth date	81.28	94.79	98.68	100.00
Name	84.10	94.79	98.55	100.00
Surname	84.10	94.79	85.84	92.31
Patronymic	84.10	94.79	98.18	100.00

Table 24.16. System evaluation (field recognition accuracy %) on MIDV-500 dataset [6].

Field group	TS, TA	KS, KA	HS, HA	PS, PA	CS, CA	Total
Numeric dates	89.56	88.46	93.96	91.21	87.36	90.11
Doc. numbers	88.54	77.08	82.29	72.92	77.08	79.58
MRZ lines	88.33	83.33	68.33	61.67	73.33	75.00
Names (Latin)	78.48	70.89	81.65	77.85	74.68	76.71

video clips with 2880 frames in total; all video clips were captured using a stationary web camera.

Table 24.16 presents the recognition accuracy of a system built with the described approaches and methods, on the video clips in MIDV-500 dataset [6]. The accuracy metric was set to be a strict string value equality: the text field recognition result was counted as correct only if its value exactly matched the ground truth (with the only pre-processing of dates format, as the analyzed system date output format is standardized). Any field that had not been returned by the system was counted as incorrect. The final accuracy was measured as the ratio of correctly recognized fields. Table 24.16 is broken down by the capturing conditions and for the whole dataset. The analyzed fields were grouped into four categories: numeric dates, document numbers, MRZ lines, and Latin name components.

The results show that the approaches and methods presented in this chapter are a viable toolkit for building identity document recognition systems which would operate on a mobile device. Though there is always room for improvement, the described methods already achieve the levels of accuracy sufficient for the application of such systems to the automatization of document data entry. The authors believe that the provided methods' overview will be a valuable resource for research in identity document analysis and recognition and will motivate future work in this field.

Acknowledgments

The authors would like to express gratitude to Igor Aleksandrovich Faradjev and Aleksandr Borisovich Merkov for their valuable comments and methodological assistance as well as to Smart Engines Service LLC for providing private experimental results.

References

[1] Aliev, M., Nikolaev, D., and Saraev, A. (2014) Construction of fast computing adjustment for niblack binarization algorithm. *Trudy ISA RAN*, 64, 25–34.

[2] Antonacopoulos, A., Bridson, D., Papadopoulos, C., and Pletschacher, S. (2009) A realistic dataset for performance evaluation of document layout analysis. *International Conference on Document Analysis and Recognition (ICDAR)* (IEEE), pp. 296–300.

[3] Arlazarov, V.L., Arlazarov, V.V., Bulatov, K.B., Chernov, T.S., Nikolaev, D.P., Polevoy, D.V., Sheshkus, A.V., Skoryukina, N.S., Slavin, O.A., and Usilin, S.A. (2022) Mobile id document recognition–coarse-to-fine approach. *Pattern Recognition and Image Analysis*, 32(1), 89–108.

[4] Arlazarov, V.V., Andreeva, E.I., Bulatov, K.B., Nikolaev, D.P., Petrova, O.O., Savelev, B.I., and Slavin, O.A. (2022). Document image analysis and recognition: A survey. *Computer Optics*, 46(4), 567–589.

[5] Arlazarov, V.V., Bulatov, K., Manzhikov, T., Slavin, O., and Janiszewski, I. (2018) Method of determining the necessary number of observations for video stream documents recognition. *Proceedings of SPIE*, 10696, 10696–10696–6.

[6] Arlazarov, V.V., Bulatov, K.B., Chernov, T.S., and Arlazarov, V.L. (2019) MIDV-500: A dataset for identity document analysis and recognition on mobile devices in video stream. *Computer Optics*, 43(5), 818–824.

[7] Arlazarov, V.V., Bulatov, K.B., and Karpenko, S.M. (2013) Recognition confidence determining method for embossed symbol recognition problem. *Proceedings of Institute for Systems Analysis RAS*, 63(3), 117–122 (in Russian).

[8] Arlazarov, V.V., Slavin, O., Uskov, A., and Janiszewski, I. (2018). Modelling the flow of character recognition results in video stream. *Bulletin of South Ural State University*, 11, 14–28.

[9] Arlazarov, V.V., Voysyat, J.S., Matalov, D.P., Nikolaev, D.P., and Usilin, S.A. (2021) Evolution of the Viola — Jones object detection method: A survey. *Bulletin of the South Ural State University Series — Mathematical Modelling, Programming & Computer Software*, 14(4), 5–23.

[10] Arlazarov, V.V., Zhukovsky, A., Krivtsov, V., Nikolaev, D., and Polevoy, D. (2014) Analysis of using stationary and mobile small-scale digital cameras for documents recognition. *Information Technologies and Computing Systems*, 3, 71–78 (in Russian).

[11] Awal, A.-M., Ghanmi, N., Sicre, R., and Furon, T. (2017) Complex document classification and localization application on identity document images. *International Conference on Document Analysis and Recognition (ICDAR)*, Vol. 01, pp. 426–431.

[12] Bay, H., Ess, A., Tuytelaars, T., and Gool, L.V. (2008) Speeded-up robust features (SURF). *Computer Vision and Image Understanding*, 110(3), 346–359. Similarity Matching in *Computer Vision and Multimedia*.

[13] Berezovsky, B.A. and Gnedin, A.V. (1984) *Problems of Best Selection* (Academia Nauk USSR, Moscow) (in Russian).

[14] Bezmaternykh, P., Ilin, D., and Nikolaev, D. (2019) U-net-bin: Hacking the document image binarization contest. *Computer Optics*, 43, 825–832. DOI: 10.18287/2412-6179-2019-43-5-825-832.

[15] Bezmaternykh, P., Nikolaev, D., and Arlazarov, V. (2018) Textual blocks rectification method based on fast Hough transform analysis in identity documents recognition. *10th International Conference on Machine Vision (ICMV 2017)*, Vol. 10696 (International Society for Optics and Photonics), p. 1069606.

[16] Bezmaternykh, P.V. and Nikolaev, D.P. (2020) A document skew detection method using fast Hough transform. In Osten, W. and Nikolaev, D.P. (eds.) *12th International Conference on Machine Vision (ICMV 2019)*, Vol. 11433 (SPIE — International Society for Optics and Photonics), pp. 132–137.

[17] Brady, M.L. (1998) A fast discrete approximation algorithm for the radon transform. *SIAM Journal on Computing*, 27(1), 107–119.

[18] Bulatov, K. (2019) A method to reduce errors of string recognition based on combination of several recognition results with per-character alternatives. *Bulletin of the South Ural State University. Series Mathematical Modelling, Programming & Computer Software*, 12(3), 74–88.

[19] Bulatov, K., Arlazarov, V., Chernov, T., Slavin, O., and Nikolaev, D. (2017) Smart IDReader: Document recognition in video stream. *14th IAPR International Conference on Document Analysis and Recognition (ICDAR)*, Vol. 6 (IEEE), pp. 39–44.

[20] Bulatov, K. and Arlazarov, V.V. (2021) Determining optimal frame processing strategies for real-time document recognition systems. *Document Analysis and Recognition – ICDAR 2021*, pp. 273–288.

[21] Bulatov, K., Fedotova, N., and Arlazarov, V.V. (2021) Fast approximate modelling of the next combination result for stopping the text recognition in a video. *2020 25th International Conference on Pattern Recognition (ICPR)*, pp. 239–246.

[22] Bulatov, K., Lynchenko, A., and Krivtsov, V. (2018) Optimal frame-by-frame result combination strategy for OCR in video stream. *Proceedings of SPIE*, 10696, 10696–10696–7.

[23] Bulatov, K., Matalov, D., and Arlazarov, V.V. (2020) MIDV-2019: Challenges of the modern mobile-based document OCR. *12th International Conference on Machine Vision (ICMV 2019)*, Vol. 11433 (SPIE — International Society for Optics and Photonics), pp. 717–722.

[24] Bulatov, K., Razumnyi, N., and Arlazarov, V.V. (2019) On optimal stopping strategies for text recognition in a video stream as an application of a monotone sequential decision model. *International Journal on Document Analysis and Recognition (IJDAR)*, 22(3), 303–314.

[25] Bulatov, K.B., Bezmaternykh, P.V., Nikolaev, D.P., and Arlazarov V.V. (2022) Towards a unified framework for identity documents analysis and recognition. *Computer Optics*, 46(3), 436–454.

[26] Bulatov, K.B., Emelyanova, E.V., Tropin, D.V., Skoryukina, N.S., Chernyshova, Y.S., Sheshkus, A.V., Usilin, S.A., Ming, Z., Burie, J., Luqman, M., and Arlazarov, V.V. (2022) Midv-2020: A comprehensive benchmark dataset for identity document analysis. *Computer Optics*, 46(2), 252–270.

[27] Bulatov, K.B., Kirsanov, V.Y., Arlazarov, V.V., Nikolaev, D.P., and Polevoy, D.V. (2016) Methods for integration the results of the documents text fields recognition in the video stream of a mobile device. *RFBR Journal*, 92(4), 109–115.

[28] Bulatov, K.B. and Polevoy, D.V. (2015) Reducing overconfidence in neural networks by dynamic variation of recognizer relevance. In Mladenov, V.M., Spasov, G., Georgieva, P., and Petrova, G. (eds.) *ECMS* (European Council for Modeling and Simulation), pp. 488–491.

[29] Burie, J.-C., Chazalon, J., Coustaty, M., Eskenazi, S., Luqman, M.M., Mehri, M., Nayef, N., Ogier, J.-M., Prum, S., and Rusiñol, M. (2015) ICDAR2015 competition on smartphone document capture and OCR (SmartDoc). *International Conference on Document Analysis and Recognition (ICDAR)* (IEEE), pp. 1161–1165.

[30] Calonder, M., Lepetit, V., Ozuysal, M., Trzcinski, T., Strecha, C., and Fua, P. (2012) Brief: Computing a local binary descriptor very fast. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 34(7), 1281–1298.

[31] Casey, R.G. and Lecolinet, E. (1996) A survey of methods and strategies in character segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 18(7), 690–706.

[32] Chazalon, J., Gomez-Krämer, P., Burie, J.-C., Coustaty, M., Eskenazi, S., Luqman, M., Nayef, N., Rusinol, M., Sidere, N., and Ogier, J.-M. (2017) SmartDoc 2017 video capture: Mobile document acquisition in video mode. *International Conference on Document Analysis and Recognition (ICDAR)*, Vol. 4 (IEEE), pp. 11–16.

[33] Chen, N. and Blostein, D. (2007) A survey of document image classification: Problem statement, classifier architecture and performance evaluation. *International Journal of Document Analysis and Recognition (IJDAR)*, 10(1), 1–16.

[34] Chernov, T., Ilin, D., Bezmaternykh, P., Faradzhev, I., and Karpenko, S. (2016) Research of segmentation methods for images of document textual blocks based on the structural analysis and machine learning. *RFBR Journal*, 4(92), 56–69.

[35] Chernov, T., Razumnuy, N., Kozharinov, A., Nikolaev, D., and Arlazarov, V. (2018). Image quality assessment for video stream recognition systems. *Proceedings of SPIE*, 10696, 10696–10696–8.

[36] Chernov, T.S., Ilyuhin, S.A., and Arlazarov, V.V. (2018) Application of dynamic saliency maps to video stream recognition systems with image quality assessment. *ICMV 2018* (SPIE).

[37] Chernov, T.S., Kolmakov, S.I., and Nikolaev, D.P. (2017) An algorithm for detection and phase estimation of protective elements periodic lattice on document image. *Pattern Recognition and Image Analysis*, 27(1), 53–65.

[38] Chernyshova, Y., Emelianova, E., Sheshkus, A., and Arlazarov, V.V. (2021) Midv-lait: A challenging dataset for recognition of ids with perso-arabic, thai, and indian scripts. *Document Analysis and Recognition – ICDAR 2021*, pp. 258–272.

[39] Chernyshova, Y.S., Gayer, A.V., and Sheshkus, A.V. (2018) Generation method of synthetic training data for mobile OCR system. *10th International Conference on Machine Vision (ICMV 2017)*, Vol. 10696 (International Society for Optics and Photonics), p. 106962G.

[40] Chow, Y., Robbins, H., and Siegmund, D. (1971) *Great Expectations: The Theory of Optimal Stopping* (Houghton Mifflin, Boston, USA).

[41] Chow, Y.S. and Robbins, H. (1961) A martingale system theorem and applications. *Proceedings of 4th Berkeley Symposium on Mathematical Statistics Probability*, Vol. 1 (University of California Press, Berkeley), pp. 93–104.

[42] Clausner, C., Antonacopoulos, A., and Pletschacher, S. (2017) ICDAR2017 competition on recognition of documents with complex layouts – RDCL2017. *International Conference on Document Analysis and Recognition (ICDAR)*, Vol. 1 (IEEE), pp. 1404–1410.

[43] Dalal, S.R. and Mallows, C.L. (1988) When should one stop testing software? *Journal of the American Statistical Association*, 83(403), 872–879.

[44] De Koker, L. (2016) Money laundering compliance the challenges of technology. *Financial Crimes: Psychological, Technological, and Ethical Issues* (Springer), pp. 329–347.

[45] Heras, de las L.-P., Terrades, O.R., Llados, J., Fernandez-Mota, D., and Canero, C. (2015) Use case visual bag-of-words techniques for camera based identity document classification. *International Conference on Document Analysis and Recognition (ICDAR)* (IEEE), pp. 721–725.

[46] Deng, J., Dong, W., Socher, R., Li, L.-J., Li, K., and Fei-Fei, L. (2009) Imagenet: A large-scale hierarchical image database. *CVPR* (IEEE), pp. 248–255.

[47] Santos, dos R.P., Clemente, G.S., Ren, T.I., and Cavalcanti, G.D. (2009) Text line segmentation based on morphology and histogram projection. *10th International Conference on Document Analysis and Recognition, 2009 (ICDAR'09)* (IEEE), pp. 651–655.

[48] Emelyanov, S., Ivanova, A., Shvets, E., and Nikolaev, D. (2018) Methods of training augmentation in the task of image classification. *Sensornye Sistemy*, 32(3), 236–245.

[49] Esser, D., Muthmann, K., and Schuster, D. (2013) Information extraction efficiency of business documents captured with smartphones and tablets. *Proceedings of the 2013 ACM Symposium on Document Engineering, DocEng'13* (ACM, New York), pp. 111–114.

[50] European Union (2016) Regulation (EU) 2016/679 of the European Parliament and of the Council of 27 April 2016 on the protection of natural persons with regard to the processing of personal data and on the free movement of such data, and repealing Directive 95/46/EC (General Data Protection Regulation). *Official Journal of the European Union*, L119, 1–88.

[51] Fan, B., Kong, Q., Trzcinski, T., Wang, Z., Pan, C., and Fua, P. (2014) Receptive fields selection for binary feature description. *IEEE Transactions on Image Processing*, 23(6), 2583–2595.

[52] Felzenszwalb, P.F. and Zabih, R. (2011) Dynamic programming and graph algorithms in computer vision. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 33(4), 721–740.

[53] Ferguson, T.S. (2008) Optimal stopping and applications. https://www.math.ucla.edu/~tom/Stopping/Contents.html. Accessed 13 November 2018.

[54] Ferguson, T.S. and Hardwick, J.P. (1989) Stopping rules for proofreading. *Journal of Applied Probability*, 26(2), 304–313.

[55] Ferguson, T.S. and Klass, M.J. (2010) House-hunting without second moments. *Sequential Analysis: Design Methods and Applications*, 29, 236–244.

[56] Fischler, M.A. and Bolles, R.C. (1981) Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography. *Communications of the ACM*, 24(6), 381–395.

[57] Fiscus, J.G. (1997) A post-processing system to yield reduced word error rates: Recognizer output voting error reduction (ROVER). *IEEE Workshop on Automatic Speech Recognition and Understanding*, pp. 347–354.

[58] Freund, Y. and Schapire, R.E. (1997) A decision-theoretic generalization of on-line learning and an application to boosting. *Journal of Computer and System Sciences*, 55(1), 119–139.

[59] Gai, K., Qiu, M., and Sun, X. (2017) A survey on fintech. *Journal of Network and Computer Applications*, 103, 262–273.

[60] Gayer, A., Ershova, D., and Arlazarov, V.V. (2023) An accurate approach to real-time machine-readable zone detection with mobile devices. *International Journal on Document Analysis and Recognition (IJDAR)*, 32(1), 1–14.

[61] Gorelik, A.L. and Skriprin, V.A. (1984) *Recognition Methods: A Textbook*, 2nd edn. (Higher Education, Moscow) (in Russian).

[62] Harley, A.W., Ufkes, A., and Derpanis, K.G. (2015) Evaluation of deep convolutional nets for document image classification and retrieval. *International Conference on Document Analysis and Recognition (ICDAR)* (IEEE), pp. 991–995.

[63] He, K., Zhang, X., Ren, S., and Sun, J. (2016) Deep residual learning for image recognition. *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 770–778.

[64] Hua, G., Liu, Z., Zhang, Z., and Wu, Y. (2006) Automatic business card scanning with a camera. *ICIP* (IEEE), pp. 373–376.

[65] Ilin, D.A. and Krivtsov, V.E. (2015) Creating training datasets for OCR in mobile device video stream. In Mladenov, V.M., Spasov, G., Georgieva, P., and Petrova, G. (eds.) *ECMS* (European Council for Modeling and Simulation), pp. 516–520.

[66] Ilyuhin, S.A., Sheshkus, A.V., and Arlazarov, V.L. (2020) Recognition of images of Korean characters using embedded networks. In Osten, W. and Nikolaev, D.P. (eds.) *12th International Conference on Machine Vision (ICMV 2019)*, Vol. 11433 (SPIE — International Society for Optics and Photonics), pp. 273–279.

[67] Jaderberg, M., Simonyan, K., Vedaldi, A., and Zisserman, A. (2016) Reading text in the wild with convolutional neural networks. *International Journal of Computer Vision*, 116(1), 1–20.

[68] Jung, K., Kim, K.I., and Jain, A.K. (2004) Text information extraction in images and video: A survey. *Pattern Recognition*, 37(5), 977–997.

[69] Kapinos, R., Feng, X., and Hilburger, P. (2015) Forming scanned composite document with optical character recognition function. US Patent App. 14/258,276.

[70] Kaur, J. and Mahajan, R. (2014) A review of degraded document image binarization techniques. *International Journal of Advanced Research in Computer and Communication Engineering*, 3(5), 6581–6586.

[71] Klass, M.J. (1973) Properties of optimal extended-valued stopping rules for s_n/n^1. *Annals of Probability*, 1(5), 719–757.

[72] Kliatskine, V., Nepomniachtchi, G., and Kotovich, N. (2014) Systems and methods for capturing critical fields from a mobile image of a credit card bill. US Patent 20140279323.

[73] Kobayashi, T., Iwamura, M., Matsuda, T., and Kise, K. (2013) An anytime algorithm for camera-based character recognition. *Proceedings of 12th International Conference on Document Analysis and Recognition*, pp. 1140–1144.

[74] Konovalenko, I.A., Shemiakina, J.A., and Faradjev, I.A. (2020) Calculation of a vanishing point by the maximum likelihood estimation method. *Bulletin of the South Ural State University. Series Mathematical Modelling, Programming & Computer Software*, 13(1), 107—117.

[75] Krasin, I., Duerig, T., Alldrin, N., Ferrari, V., Abu-El-Haija, S., Kuznetsova, A., Rom, H., Uijlings, J., Popov, S., Kamali, S., Malloci, M., Pont-Tuset, J., Veit, A., Belongie, S., Gomes, V., Gupta, A., Sun, C., Chechik, G., Cai, D., Feng, Z., Narayanan, D., and Murphy, K. (2017) OpenImages: A public dataset for large-scale multi-label and multi-class image classification. github.com/openimages.

[76] Krizhevsky, A., Sutskever, I., and Hinton, G.E. (2012) Imagenet classification with deep convolutional neural networks. In Pereira, F., Burges, C., Bottou, L., and Weinberger, K. (eds.) *Advances in Neural Information Processing Systems*, Vol. 25 (Curran Associates, Inc.), pp. 1097–1105.

[77] Kryuchkova, K., Sheshkus, A., and Bulatov, K. (2022) Neural network-based prediction of the stopping moment for text recognition in a video stream. *2022 26th International Conference on Pattern Recognition (ICPR)*, pp. 1450–1456.

[78] Kumar, J., Ye, P., and Doermann, D. (2013) A dataset for quality assessment of camera captured document images. *Camera Based Document Analysis and Recognition (CBDAR)* (Springer), pp. 113–125.

[79] LeCun, Y. (1998) The MNIST database of handwritten digits. http://yann.lecun.com/exdb/mnist/.

[80] LeCun, Y., Bottou, L., Bengio, Y., and Haffner, P. (1998) Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11), 2278–2324.

[81] Lepetit, V. and Fua, P. (2004) Towards recognizing feature points using classification trees. Technical report.

[82] Limonova, E., Bezmaternykh, P., Nikolaev, D., and Arlazarov, V. (2017) Slant rectification in Russian passport OCR system using fast Hough transform. *Proceedings of SPIE*, 10341, 103410P–103410P–5.

[83] Limonova, E., Matveev, D., Nikolaev, D., and Arlazarov, V.V. (2020) Bipolar morphological neural networks: Convolution without multiplication. In Osten, W. and Nikolaev, D.P. (eds.) *12th International Conference on Machine Vision (ICMV 2019)*, Vol. 11433 (SPIE — International Society for Optics and Photonics), pp. 962–969.

[84] Limonova, E.E., Alfonso, D.M., Nikolaev, D.P., and Arlazarov, V.V. (2021) Bipolar morphological neural networks: Gate-efficient architecture for computer vision. *IEEE Access*, 9, 97569–97581.

[85] Limonova, E.E., Terekhin, A.P., Nikolaev, D.P., and Arlazarov, V.V. (2016) Fast implementation of morphological filtering using arm neon extension. *IJAER*, 11(24), 11675–11680.

[86] Lowe, D.G. (2004) Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision*, 60(2), 91–110.

[87] Luqman, M.M., Gomez-Krämer, P., and Ogier, J.-M. (2014) *Mobile Phone Camera-Based Video Scanning of Paper Documents* (Springer International Publishing, Cham), pp. 164–178.

[88] Marinai, S., Gori, M., and Soda, G. (2005) Artificial neural networks for document analysis and recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 27(1), 23–35.

[89] Matalov, D., Limonova, E., Skoryukina, N., and Arlazarov, V.V. (2021) Rfdoc: Memory efficient local descriptors for id documents localization and classification. *Document Analysis and Recognition — ICDAR 2021*, pp. 209–224.

[90] Matalov, D.P., Limonova, E.E., Skoryukina, N.S., and Arlazarov, V.V. (2023) Memory efficient local features descriptor for identity document detection on mobile and embedded devices. *IEEE Access*, 11, 1104–1114.

[91] Matalov, D.P., Usilin, S.A., and Arlazarov, V.V. (2020) Single-sample augmentation framework for training Viola-Jones classifiers. *12th International Conference on Machine Vision (ICMV 2019)*, Vol. 11433 (SPIE - International Society for Optics and Photonics), pp. 124–131.

[92] Minkina, A., Nikolaev, D., Usilin, S., and Kozyrev, V. (2015) Generalization of the Viola-Jones method as a decision tree of strong classifiers for real-time object recognition in video stream. *Proceedings of SPIE*, 9445, 944517–944517–5.

[93] Mucci, A.G. (1973) On a class of secretary problems. *Annals of Probability*, 1(3), 417–427.

[94] Muja, M. and Lowe, D.G. (2012) Fast matching of binary features. *2012 9th Conference on Computer and Robot Vision (CRV)* (IEEE), pp. 404–410.

[95] Nagy, G. (2000) Twenty years of document image analysis in PAMI. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(1), 38–62.

[96] Nayef, N., Luqman, M.M., Prum, S., Eskenazi, S., Chazalon, J., and Ogier, J.-M. (2015) SmartDoc-QA: A dataset for quality assessment of smartphone captured document images-single and multiple distortions. *International Conference on Document Analysis and Recognition (ICDAR)* (IEEE), pp. 1231–1235.

[97] Nikolaev, D., Polevoy, D., and Tarasova, N. (2014) Training data synthesis in text recognition problem solved in threedimensional space. *Information Technologies and Computer Systems*, 3, 82–88.

[98] Nikolaev, D.P., Karpenko, S.M., Nikolaev, I.P., and Nikolayev, P.P. (2008) Hough transform: Underestimated tool in the computer vision field. *Proceedings of the 22th European Conference on Modelling and Simulation*, Vol. 238, p. 246.

[99] Otsu, N. (1979) A threshold selection method from gray-level histograms. *IEEE Transactions on Systems, Man, and Cybernetics*, 9(1), 62–66.

[100] Papageorgiou, C.P., Oren, M., and Poggio, T. (1998) A general framework for object detection. *6th International Conference on Computer Vision, 1998* (IEEE), pp. 555–562.

[101] Pereyra, G., Tucker, G., Chorowski, J., Kaiser, Ł., and Hinton, G. (2017) Regularizing neural networks by penalizing confident output distributions. *ArXiv e-prints* arXiv:1701.06548.

[102] Petrova, O., Bulatov, K., and Arlazarov, V.L. (2020) Methods of weighted combination for text field recognition in a video stream. *12th International Conference on Machine Vision (ICMV 2019)*, Vol. 11433 (SPIE - International Society for Optics and Photonics), pp. 704–709.

[103] Petrova, O.O., Bulatov, K.B., Arlazarov, V.V., and Arlazarov, V.L. (2021) Weighted combination of per-frame recognition results for text recognition in a video stream. *Computer Optics*, 45(1), 77–89.

[104] Polevoy, D., Bulatov, K., Skoryukina, N., Chernov, T., Arlazarov, V., and Sheshkus, A. (2016) Key aspects of document recognition using small digital cameras. *RFBR Journal*, 4(92), 98–105.

[105] Polevoy, D.V., Sigareva, I.V., Ershova, D.M., Arlazarov, V.V., Nikolaev, D.P., Ming, Z., Luqman, M.M., and Burie, J.-C. (2022) Document liveness challenge dataset (DLC-2021). *Journal of Imaging*, 8(7), 1–12.

[106] Pratikakis, I., Zagoris, K., Barlas, G., and Gatos, B. (2017) ICDAR2017 competition on document image binarization (DIBCO 2017). *International Conference on Document Analysis and Recognition (ICDAR)*, Vol. 1 (IEEE), pp. 1395–1403.

[107] Ronneberger, O., Fischer, P., and Brox, T. (2015) U-net: Convolutional networks for biomedical image segmentation. In Navab, N., Hornegger, J., Wells, W.M., and Frangi, A.F. (eds.) *Medical Image Computing and Computer-Assisted Intervention — MICCAI 2015* (Springer International Publishing, Cham), pp. 234–241.

[108] Rybakova, E.O., Trusov, A.V., Limonova, E.E., Skoryukina, N.S., Bulatov, K.B., and Nikolaev, D.P. (2023) PESAC, the generalized framework for RANSAC-based methods on simd computing platforms. *IEEE Access*, 11, 82151–82166.

[109] Saba, T., Sulong, G., and Rehman, A. (2010) A survey on methods and strategies on touched characters segmentation. *International Journal of Research and Reviews in Computer Science*, 1(2), 103–114.

[110] Shemiakina, J., Limonova, E., Skoryukina, N., Arlazarov, V.V., and Nikolaev, D.P. (2021) A method of image quality assessment for text recognition on camera-captured and projectively distorted documents. *Mathematics*, 9(17), 1–22.

[111] Shemiakina, J., Zhukovsky, A., and Faradjev, I. (2017) The calculation of a projective transformation in the problem of planar object targeting by feature points, pp. 103411N–103411N–6.

[112] Sher, A., Trusov, A., Limonova, E., Nikolaev, D., and Arlazarov, V.V. (2023) Neuron-by-neuron quantization for efficient low-bit QNN training. *Mathematics*, 11(9), 1–17.

[113] Sheshkus, A. and Arlazarov, V.L. (2015) Space symbol detection on complex background using visual context. *ECMS*, pp. 532–536.

[114] Sheshkus, A., Ingacheva, A., Arlazarov, V., and Nikolaev, D. (2019) Houghnet: Neural network architecture for vanishing points detection. *2019 International Conference on Document Analysis and Recognition (ICDAR)*, pp. 844–849.

[115] Sidere, N., Cruz, F., Coustaty, M., and Ogier, J.-M. (2017) A dataset for forgery detection and spotting in document images. *Emerging Security Technologies* (IEEE), pp. 26–31.

[116] Simon, M., Rodner, E., and Denzler, J. (2015) Fine-grained classification of identity document types with only one example. *MVA* (IEEE), pp. 126–129.

[117] Simonyan, K. and Zisserman, A. (2014) Very deep convolutional networks for large-scale image recognition. *ArXiv preprint* arXiv:1409.1556.

[118] Skoryukina, N., Arlazarov, V., and Nikolaev, D. (2019) Fast method of id documents location and type identification for mobile and server application. *2019 International Conference on Document Analysis and Recognition (ICDAR)*, pp. 850–857.

[119] Skoryukina, N., Faradjev, I., Bulatov, K., and Arlazarov, V.V. (2020) Impact of geometrical restrictions in RANSAC sampling on the ID document classification. In Osten, W. and Nikolaev, D.P. (eds.) *12th International Conference on Machine Vision (ICMV 2019)*, Vol. 11433 (SPIE — International Society for Optics and Photonics), pp. 35–41.

[120] Skoryukina, N., Nikolaev, D.P., Sheshkus, A., and Polevoy, D. (2015) Real time rectangular document detection on mobile devices. *Proceedings of SPIE*, 9445, 94452A–94452A–6.

[121] Slavin, O.A. (2011) Adaptive recognition and its application for printed text entry system: Dr. sc. dissertation (in Russian).

[122] Slugin, D.G. and Arlazarov, V.V. (2017) Text fields extraction based on image processing. *Trudy ISA RAN*, 67(4), 65–73.

[123] Smith, R. (2007) An overview of the Tesseract OCR engine. *Proceedings of the 9th International Conference on Document Analysis and Recognition - Volume 02, ICDAR'07* (IEEE Computer Society, Washington), pp. 629–633.

[124] Stuner, B., Chatelain, C., and Paquet, T. (2017) LV-ROVER: Lexicon verified recognizer output voting error reduction. *ArXiv e-prints* arXiv:1707.07432.

[125] Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S., Anguelov, D., Erhan, D., Vanhoucke, V., and Rabinovich, A. (2015) Going deeper with convolutions. *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 1–9.

[126] Taigman, Y., Yang, M., Ranzato, M., and Wolf, L. (2014) Deepface: Closing the gap to human-level performance in face verification. *2014 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1701–1708.

[127] Tamaki, M. (2015) On the optimal stopping problems with monotone thresholds. *Journal of Applied Probability*, 52(4), 926–940.

[128] Tropin, D.V., Ilyuhin, S.A., Nikolaev, D.P., and Arlazarov, V.V. (2021) Approach for document detection by contours and contrasts. *2020 25th International Conference on Pattern Recognition (ICPR)*, pp. 9689–9695.

[129] Trusov, A., Limonova, E., Nikolaev, D., and Arlazarov, V.V. (2022) Fast matrix multiplication for binary and ternary CNNs on ARM CPU. *2022 26th International Conference on Pattern Recognition (ICPR)*, pp. 3176–3182.

[130] Trusov, A., Limonova, E., Slugin, D., Nikolaev, D., and Arlazarov, V.V. (2021) Fast implementation of 4-bit convolutional neural networks for mobile devices. *2020 25th International Conference on Pattern Recognition (ICPR)*, pp. 9897–9903.

[131] Trusov, A.V., Limonova, E.E., and Arlazarov, V.V. (2023) Fast computation of RFD-like descriptors in four orientations. *IEEE Access*, 11, 19725–19740.

[132] Trusov, A.V., Limonova, E.E., Nikolaev, D.P., and Arlazarov, V.V. (2021). p-im2col: Simple yet efficient convolution algorithm with flexibly controlled memory overhead. *IEEE Access*, 9, 168162–168184.

[133] Usilin, S., Nikolaev, D., Postnikov, V., and Schaefer, G. (2010) Visual appearance based document image classification. *2010 IEEE International Conference on Image Processing*, pp. 2133–2136.

[134] Van Der Zijp, J. (2008) Fast half float conversions. Technical report. ftp://www.fox-toolkit.org/pub/fasthalffloatconversion.pdf.

[135] Van Herk, M. (1992) A fast algorithm for local minimum and maximum filters on rectangular and octagonal kernels. *Pattern Recognition Letters*, 13(7), 517–521.

[136] Veit, A., Matera, T., Neumann, L., Matas, J., and Belongie, S. (2016) Coco-text: Dataset and benchmark for text detection and recognition in natural images. *Arxiv preprint* arXiv:1601.07140.

[137] Viola, P. and Jones, M. (2001) Rapid object detection using a boosted cascade of simple features. *Proceedings of the 2001 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2001. CVPR 2001*, Vol. 1 (IEEE), pp. I–I.

[138] Viola, P. and Jones, M.J. (2004) Robust real-time face detection. *International Journal of Computer Vision*, 57(2), 137–154.

[139] Wang, X., Bissacco, A., Berntson, G., Nazif, M., Scheiner, J., Shih, S., Snyder, M., and Talavera, D. (2014) Client side filtering of card ocr images. US Patent 8903136.

[140] Weiss, G.M. and Provost, F. (2001) The effect of class distribution on classifier learning: An empirical study. Rutgers University.

[141] Wemhoener, D., Yalniz, I.Z., and Manmatha, R. (2013) Creating an improved version using noisy OCR from multiple editions. *Proceedings of 12th International Conference on Document Analysis and Recognition, ICDAR'13* (IEEE Computer Society), pp. 160–164.

[142] Wu, J.-C., Hsieh, J.-W., and Chen, Y.-S. (2008) Morphology-based text line extraction. *Machine Vision and Applications*, 19(3), 195–207.

[143] Yang, M.C.K., Wackerly, D.D., and Rosalsky, A. (1982) Optimal stopping rules in proofreading. *Journal of Applied Probability*, 19(3), 723–729.

[144] Yujian, L. and Bo, L. (2007) A normalized Levenshtein distance metric. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 29(6), 1091–1095.

[145] Zeiler, M.D. and Fergus, R. (2014) Visualizing and understanding convolutional networks. In Fleet, D., Pajdla, T., Schiele, B., and Tuytelaars, T. (eds.) *Computer Vision – ECCV 2014* (Springer International Publishing, Cham), pp. 818–833.

[146] Zhang, Y., Gueguen, L., Zharkov, I., Zhang, P., Seifert, K., and Kadlec, B. (2017) Uber-text: A large-scale dataset for optical character recognition from street-level imagery. *SUNw: Scene Understanding Workshop-CVPR*.

[147] Zhukovskiy, A.E., Nikolaev, D.P., Arlazarov, V.V., Postnikov, V.V., Polevoy, D.V., Skoryukina, N.S., Chernov, T.S., Shemyakina, Y.A., Mukovozov, A.A., Konovalenko, I.A., and Povolotskiy, M.A. (2018) Segments graph-based approach for document capture in a smartphone video stream. *ICDAR 2017*, Vol. 1 (Conference Publishing Services (CPS)), pp. 337–342.

[148] Zilberstein, S. (1996) Using anytime algorithms in intelligent systems. *AI Magazine*, 17, 73–83.

© 2025 World Scientific Publishing Company
https://doi.org/10.1142/9789811267215_0025

Chapter 25

Multilayer Constructions of the Algebraic Approach in Pattern Recognition

Yury I. Zhuravlev, **Konstantin V. Rudakov**, and **Yury G. Smetanin**

Federal Research Center "Computer Science and Control" of the Russian Academy of Sciences, Moscow, the Russian Federation

This chapter describes a general framework of multilayer constructions of algebraic approach to the problem of synthesizing pattern recognition algorithms. The approach gives a unified mathematical description of pattern recognition methods including the most sophisticated ones, e.g. deep neural networks. It provides easy to verify criteria for the correct solvability of problem instances. Algebraic approach is a universal tool for solving pattern recognition problems, including those with relatively small training sets. A principal feature of algebraic approach is that it considers basic families of algorithms as sets of elements of an algebra over algorithms, the algebraic operations are used to construct new algorithms that we need. Within the approach, it is possible to check, for a given instance, whether the problem is well posed and whether the input information is consistent and sufficient for obtaining a correct (in the formally defined sense) solution. If the easily verified conditions of correctness and completeness are satisfied, the approach makes it possible to obtain a correct solution to the pattern recognition

problem. Input data for the implementation are the values of both the training set and the set to be recognized. Basic algorithms construct the estimations of the objects to be recognized (these estimations may be interpreted as new features obtained by calculations), and algebraic operations are over these estimations. At the last step, the classification of the objects is obtained using the estimations. So, the input features are usually the observable or measured values, and algebraic operations over-estimations may be interpreted as generating new features at a higher level of abstraction. Difficulties that appear in using algebraic approach are the same as in other universal methods: hardness and the necessity of elaborate learning given an instance. Multilayer constructions are means to eliminate these difficulties by decomposing hard problems that replace them with sets of simpler problems that are also convenient for parallel implementations.

Keywords: Multilayer construction, algebraic approach, pattern recognition, heuristic models, correcting operations.

25.1 Introduction

The so-called algebraic approach to the problem of synthesizing correct algorithms based on heuristic models appeared in the seventies of the last century [16–19]. In the previous period, the standard approach was as follows: heuristic algorithms were proposed and tested for certain application problems of pattern recognition, and sometimes it was possible to obtain results with acceptable quality. The most successful algorithms turned into heuristic models of algorithms, which were actually parametric families of mappings of input descriptions into the desired answers.

It should be noted that the considered scientific and technological direction has acquired a lot of modifications over the past years/including classification, clustering, forecasting, data mining, machine learning, neural networks, decision-making, etc. What is similar for all of them is that the solution to the problem is based not on an adequate mathematical model of the domain, but on modeling the process of information analysis, that is, converting the input data into the answers to the problem.

Note that multilayer nature was inherent to algorithmic constructions from the very beginning: for example, to obtain answers about the belonging of objects to classes, some estimates of the

corresponding "probabilities" were first calculated (though these estimates were not compared to any explicit Borel sets as measures), and then these "probabilities" were transformed into "yes-no" answers. The process of using the outputs of the algorithms of the previous layer as inputs for the algorithms of the next layer can be interpreted as an operation on the algorithms. The formalization and development of this idea led to the creation of the considered algebraic approach, in which the algorithms are interpreted as points of the corresponding functional space, and the results of algebraic operations on them are studied.

In the development of the algebraic approach, the corresponding conceptual apparatus was formed, and a brief description of it is the aim of this chapter. It seems to us that it may also be useful for analyzing the latest methods of analysis and solving problems in this field.

25.2 On Problems, Admissible Mappings, and Correct Algorithms

The algebraic approach to the problem of correct algorithm synthesis was originally proposed for the problems of pattern recognition or classification. However, the general principles underlying the approach can be presented at a much more general level, omitting unnecessary details.

So, we will assume that two non-empty sets are initially defined, hereinafter they are called the space of initial information $\Im_i = \{I_i\}$ and the space of final information (answers) $\Im_f = \{I_f\}$. The problem is to build an algorithm that implements a mapping from the first space to the second one. In the original version of the algebraic approach [16–19], the spaces[a] $\mathsf{C}_{ql}(U)$ of matrices over arbitrary multi-element sets U, called information matrices and information matrices, were studied as spaces of initial and final information, respectively. The "physical sense" of these matrices is as follows:

- the set of objects $\mathsf{S} = \{S\}$ to be recognized or classified is to be considered,

[a] The space of $\mathbf{q} \times l$-matrices over an arbitrary set U will be denoted as $\mathsf{C}_{ql}(U)$.

- it is assumed that the set I_{ob} of descriptions of objects is defined (as a rule, I_{ob} is the set of vectors containing the values of the object features),
- it is also assumed that there is a function $D_{\text{ob}} : \mathsf{S} \to \mathsf{I}_{\text{ob}}$ that associates each object S from S with its description $D_{\text{ob}}(S)$ (in many model cases it makes no sense to distinguish objects and their descriptions, say, in the case, the objects themselves are points in n-dimensional Euclidean space),
- the task is to assign objects to classes, so it is assumed that there is a set $K_1, \ldots, K_l$ of subsets of the set S, called classes,
- further, there is a natural assumption that there are many descriptions of the classes I_{cl} and the function: $\mathsf{D}_{\text{cl}} : 2^S \to I_{\text{cl}}$ (the description of a class K_j often includes information about the belonging of m objects from some training subsample $(S_1, \ldots S_m) \in S$ to this class; for example, it can be a Boolean n-vector;,
- in many cases, the elements of the information matrix $\hat{I} = \|I_{ij}\|_{q\times l}$ are pairs:

$$I_{ij} = (D_{\text{ob}}(S_i), D_{\text{cl}}(K_j)),$$

  in the general case, it is assumed that information matrices are simply matrices of dimension $q \times l$ over some set I containing more than one element;,
- finally, it is assumed that the set $\tilde{\mathsf{I}}$ is given of possible answers to questions of whether the objects belong to the classes (often$\tilde{\mathsf{I}} = (0,\ 1)$, and then the information matrix $\hat{\tilde{I}} \in \mathsf{C}_{ql}(\tilde{\mathsf{I}})$ turns out to be a matrix whose elements belong to this set.

Meaningful problems appear as problems of constructing algorithms that implement mappings from initial information to final information satisfying certain conditions. We assume that these conditions distinguish the subset of so-called admissible mappings in the set $M*$ of all mappings f from I_i to I_f. Any algorithm A that implements any of the admissible mappings f is called the correct one and is considered to be the solution to the problem.

Following the tradition of the algebraic approach, the tasks of constructing correct algorithms will be called problems Z, and the set of admissible mappings will be denoted by $M(Z)$ (of course, $M(Z)$ is a subset of the set M^*).

The requirements defining the set of admissible mappings $M(Z)$ will be denoted by I_{str} and called the structural information in the problem Z, assuming that $M(Z) = M(I_{\text{str}})$.

The extremely typical part of structural information is usually the information $I_{\text{str}}^{\text{pr}}$ about precedents from the test sample $S_1, \ldots, S_q$, which determines the requirement for admissible mappings, expressed by the equality $f(\hat{I}) = \hat{\tilde{I}}$, so that in this case

$$\mathsf{M}(I_{\text{str}}^{\text{pr}}) = \{f | f \in \mathsf{M}^*, f(\hat{I}) = \hat{\tilde{I}}\}.$$

Note that we can use the precedents in a "softer" version, assuming, for instance, that

$$\mathsf{M}(I_{\text{str}}^{\text{pr}}) = \{f | f \in \mathsf{M}^*, r(f(\hat{I}), \hat{\tilde{I}}) \leq \varepsilon \cdot q \cdot l\},$$

where the metric r that we use is, for example, the Hamming metrics on binary matrices.

Obviously, it is impossible to limit oneself to the "precedent" case: the task in this case immediately admits a trivial solution — a constant mapping whose output is an a priori known information matrix of the problem Z for arbitrary inputs. Hence, it is necessary to consider and take into account additional "precision" on precedents, the requirements for permissible mappings expressed by the "universal" part of structural information $\hat{I}_{\text{str}}^{u}$. Considering further the process of calculating the values of functions from M as constructing superpositions of mappings from spaces X to another spaces Y, we assume that $\hat{I}_{\text{str}}^{u}$ determines ("cuts out") a corresponding subset of mappings determined in each set of such mappings ("cuts out"). Thus, the formalism used in the algebraic approach appears, in which

$$\mathsf{M}(I_{\text{str}})(\mathsf{I}_i, \mathsf{I}_f) = \mathsf{M}(I_{\text{str}}^{\text{pr}})(\mathsf{I}_i, \mathsf{I}_f) \cap \mathsf{M}(I_{\text{str}}^{u})(\mathsf{I}_i, \mathsf{I}_f).$$

So, we can exactly describe the conditions of solvability for the problem Z as

$$\mathsf{M}(I_{\text{str}}^{\text{pr}})(\mathsf{I}_i, \mathsf{I}_f) = \mathsf{M}(I_{\text{str}}^{\text{pr}})(\mathsf{I}_i, \mathsf{I}_f) \cap \mathsf{M}(I_{\text{str}}^{u})(\mathsf{I}_i, \mathsf{I}_f) \neq,$$

essentially expressing the requirement of consistency of precedent and universal information.

As an example, consider a problem in which the classification of an object should not depend on the corresponding row number

in the information matrix. It is natural to solve such a problem by defining admissible mappings as those commuting with arbitrary row substitutions in matrices of information and information matrices. Requirements of this kind are described by categories of a special type, and the sets $\mathsf{M}(I^u_{\text{str}})$are defined as sets of morphisms of these categories [6, 7].

Next example: A situation when admissible mappings should be monotone. It is assumed, of course, that there are relations of order I_i and I_f, formally describing "qualitative" knowledge about the subject area ("the stronger the inflammation, the higher the temperature of the patient," etc.). There is an interesting, but little studied case where it is possible to introduce a substantively justified relation "between", and the admissible mappings must satisfy, for example, the following requirement: if two objects belong to the same class, all objects lying between them must also be assigned to this class. Such universal constraints lead to the description of $\mathsf{M}(I^u_{\text{str}})$ as the sets of homomorphisms of the corresponding relational systems [13, 14].

In addition to precedents requirements for admissible mappings, so-called set-theoretic constraints were also studied [8,9]. In this case, for each point in the space of initial information I_i, its own range of admissible values is defined in the space of final information — a subset of the space I_f; the mappings from I_i to I_f are admissible if and only if the image of each point belongs to the region of admissible values defined for this point.

Let us consider problems Z from the standpoint of Hadamard correctness. Within the framework of the initial constructions of the algebraic approach, the question of solvability was mainly investigated. The question of the uniqueness of solutions almost always has a negative answer in different senses. First, the singletonity of the set $\mathsf{M}(I_{\text{str}})$ means that we have absolutely complete information about the real situation, which is almost never encountered in practice. At the intuitive level, it is clear that as the available information on the subject area increases, the set of admissible mappings $\mathsf{M}(I_{\text{str}})$ should narrow down. Second, we consider the algorithm to be the solution if it implements an admissible mapping, and, generally speaking, this mapping can be implemented by essentially different algorithms. In our review, this question is not considered in detail, so we will no longer, as a rule, consider the difference between algorithms and the maps they implement. Finally, we note that the question of stability

of solutions requires, for its consideration, to involve at least topological structures in the spaces under consideration. In a sense, the question of stability in the framework of the algebraic approach is reflected in the concept of regular problems and in the study of the so-called radii of solvability and regularity [8, 9].

In order to introduce the notion of a regular problem in the most general form, assume that a certain set Z of problems is fixed and that for each problem Z in Z, a "neighborhood" is defined in Z, i.e. a subset $O(Z) \subseteq \mathsf{Z}$ such that $Z \in O(Z)$. A problem Z is called regular if all the problems from its neighborhood $O(Z)$ are solvable. Note that the choice of a system of neighborhoods actually turns out to be a parameter for the notion of regularity. In particular, if a neighborhood of each problem consists only of itself, then the original notion of solvability is a particular case of regularity.

When considering classification problems for the definition of the concept of regular problems Z, we study many problems that have common structural information $\hat{I}^u_{\text{str}}$and differ only in pairs of matrices$(\hat{I}, \hat{\tilde{I}})$. In the original papers [17–19,21], the case was considered mainly when a neighborhood of a problem with a given information matrix was assumed to be the set of problems with the same information matrix $\hat{I}$ and arbitrary answer information matrices $\hat{\tilde{I}}$ (Zhuravlev regularity).

25.3 On Estimation Spaces, Algorithm Models, and Correcting Operations

Designing algorithms that implement admissible mappings reduces, as a rule, to solving some problems of conditional optimization. In the simplest case, we can assume that initially there is some parametric family $\mathsf{M}(\pi)$ of mappings from I_i to I_f, and we need to choose a parameter value π such that the corresponding mapping f from $\mathsf{M}(\pi)$ belongs to the set $\mathsf{M}(I_{\text{str}})$. Note that the parameter space itself can be quite complex, including, say, set-theoretic components (see, for example, the case of estimation calculation algorithms described in the following).

The most important circumstance, which largely determines the distinction from the classical mathematical modeling, is the heuristic

nature of parametric families $\mathsf{M}(\pi)$, which are called "models of algorithms" in our case, in contrast to the mathematical models of subject areas where traditional approaches are based. To emphasize this difference, the term "heuristic information models" is often used, emphasizing the fact that they reflect not the domain model, but only some considerations and conjectures about what the interrelationships of the initial data and the desired answers can be. We can say that it is not the domain that is being modeled, but the process of transforming input information into output information.

The heuristic nature of algorithm models justifies a fundamental for the algebraic approach way to use these models: not just as areas within which optimization problems are solved (in other words, identification of parameter values, parameter settings, training, etc.), but as "warehouses of building material", on the basis of which it is possible to purposefully build new algorithms and require just them to be correct, i.e. implement valid mappings.

Building the elements of a certain set of elements on the basis of the elements of the same set is described naturally as an operation on this set, so in our case, we are talking about algebraic systems [1], in which the arguments of the operations are mappings as elements of the corresponding functional space. At the general formal level, assuming that there is a heuristic model of algorithms $\mathsf{M}(\pi)$ and a family of operations F, we can say that in the algebraic approach, the construction of solutions within the framework $\mathsf{M}(\pi)$ is replaced by the search for solutions in the family

$$\mathsf{F}(\mathsf{M}(\pi)) = \{F(f_1, \ldots, f_n) | F \in \mathsf{F}, (f_1, \ldots, f_n) \in \mathsf{M}^{\mathsf{n}}(\pi)\},$$

called the F-extension of the algorithm model $\mathsf{M}(\pi)$. Note that the inclusion $\mathsf{M}(\pi) \in \mathsf{F}(\mathsf{M}(\pi))$ is easily ensured by the requirement that the family F includes the unary operator of identity.

In the algebraic approach, operators from F are called correcting operations. On the intuitive level, this term reflects the idea of compensating for errors and other shortcomings of some algorithms at the expense of others when used together, which goes back, in particular, to the results of C. Shannon about building reliable sensors from unreliable [11, 12] and the work of many other authors. Among the methods based on using initial algorithms together, we also mention the "committee method" of Mazurov [2, 3] and "areas of competence" of Rasstrigin [5].

Correcting operations are defined as operations over mappings. These operations ate induces usually by operations over the areas of values of mappings' arguments: given n functions $f_1, \ldots, f_n$ from the set X into the set Y and the operation F over the set Y, that is, the mapping $F : Y^n \to Y$, we can define the operation F^* over the set M of all mappings from X into Y as a function F^* from M^n into M, using the definition

$$F^*(f_1, \ldots, f_n)(x) = F(f_1(x), \ldots, f_{nj}(x))$$

for all x from X. That is just the method for defining the operations over mappings applied in the described algebraic approach. For its formal description, traditional products and diagonalizations of mappings are used:

if functions

$$f_1 : X_1 \to Y_1 \quad \text{and} \quad f_2 : X_2 \to Y_2$$

are given, then the product function

$$f_1 \times f_2 : X_1 \times X_2 \to Y_1 \times Y_2$$

is defined such that

$$f_1 \times f_2(x_1, x_2) = (f_1(x_1), f_2(x_2))$$

for all (x_1, x_2) from $X_1 \times X_2$;

if a function

$$f : X^n \to Y$$

is given, then the diagonalization function

$$f_\Delta : X \to Y$$

is defined such that

$$f_\Delta(x) = f(x, \ldots, x)$$

for all x from X. Now we can describe the correcting operation F^* in the following way:

$$F^*(f_1, \ldots, f_n) = F \circ (f_1 \times f_2 \times \cdots \times f_n)_\Delta.$$

What is significant here is that the operation on maps turns out to be a superposition of "ordinary" functions. Note that

- $(f_1 \times f_2 \times \cdots \times f_n)$ is a function from X^n into Y^n,
- diagonalization $(f_1 \times f_2 \times \cdots \times f_n)_\Delta$ is a function form X into Y^n, and
- operation F is a function from Y^n into Y,

so the superposition $F_0(f_1 \times f_2 \times \cdots \times f_n)_\Delta$ is defined correctly. Note that the image of the element x from the initial support of X under the mapping $(f_1 \times f_2 \times \cdots \times f_n)$ is the set of images of the functions $f_1, \ldots, f_n$.

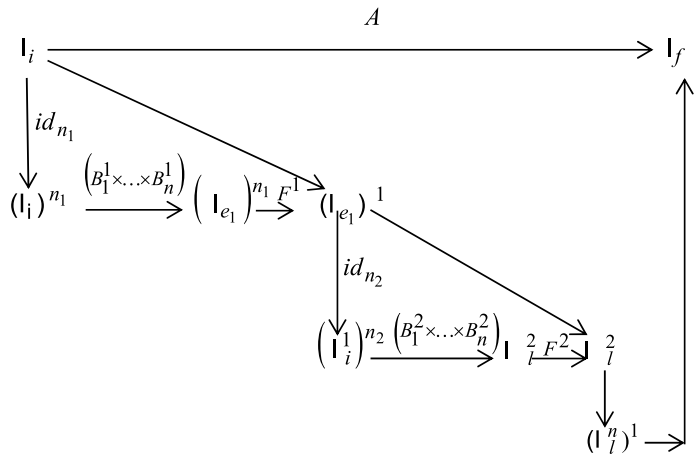
The direct construction of correcting operations on the basis of operations on a set of values of mappings is in most practical cases an insufficiently convenient technique. The reason is that the space of the final information I_f is determined by the meaningful requirements for the solution of a specific applied problem. For example, in the classification problems, the "answer" for an object can be a Boolean vector whose length is determined by the number of simultaneously considered classes, so the information matrix also turns out to be Boolean. In [16], a three-valued case is considered as a negative example. In general, we can say that the space I_f determined by the substantive requirements usually turns out a "test stand" for defining correcting operations, not sufficiently convenient from the technical point of view.

The idea exploited in the algebraic approach is based on using spaces I_e, which are "intermediate" in relation to I_i and I_f; they are called evaluation spaces. In fact, the calculation of some scalar numerical estimates and their subsequent comparison with thresholds as a technique was widely used and is still used in a wide scope of variants. From our point of view, it is essential that the algorithms are constructed as superpositions, where the outputs of some algorithms become inputs of the "next layer" algorithms, and the spaces of estimations are chosen to a large extent from the point of "technical convenience" (in the mathematical sense of the word), which makes it natural to use operations on them as sources of correcting operations over heuristic algorithms.

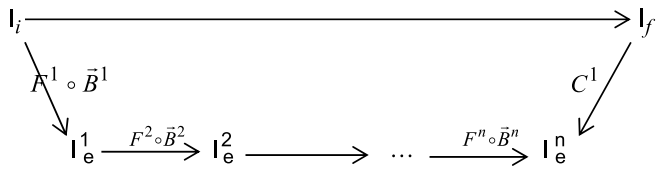
So, the process of calculating, by the desired algorithm, the values of an admissible mapping at some point in space I_i will be considered

as a sequential computation of the values of mappings B from the initial information space to the estimation spaces, the mappings B from some evaluation spaces to the "next layer" spaces (these maps are traditionally [15] called recognition or algorithmic operators), and, finally, from spaces of the "penultimate layer" into the answer space. In this particular case, the space of the "penultimate layer" may coincide with the answer space and the "last mapping" C, usually called the "decision rule" in the algebraic approach, may even be identical, which is sometimes useful to keep in mind for the unification of formalism.

At the general level, the above can be illustrated by the following commutative diagram (here id_k is the injective embedding of the set X into its kth Cartesian degree, that is $id_k : X \to X^k, id_k(x) = (x, \ldots, x)$:



It is fundamentally important here that in the process of calculating the value of the admissible mapping f, the sequential computation of the "multi-images" of initial information (element of the space I_i) is done in the spaces of estimation:



As noted above, estimation spaces are in a sense "virtual": they are introduced and used only "inside" the computational process. Transitions between them can in many cases be considered as the addition of new calculated features of objects or phenomena and the elimination of some of the "old" features. The main thing here is the fact that during the next transition to I_e^k, all the information available for further use is contained in the corresponding "multi-image" in $(\mathsf{I}_e^k)^{n_k}$. An element of space $(\mathsf{I}_e^k)^{n_k}$ can be, say, a set of neuron outputs of a certain layer of an artificial neural network or a set of estimation matrices generated by the recognition operators of the model of estimation algorithms [20]. A description of one of the original variants of this model will be our immediate goal.

We consider classification problems in which the initial descriptions of objects S belong to the Cartesian product $\mathsf{S} = M_1 \times M_2 \times \cdots \times M_n$ of sets, on Cartesian squares of which nonnegative functionals $\rho_1, \ldots, \rho_n$ are defined, which are usually called "distances" or "semimetrics". In this case, it is only required that the condition $\rho_t(S, S) = 0$ be satisfied for all t from $\{1, \ldots, n\}$ and for all S.

The sets I_{ob} of descriptions of objects and I_{cl} of descriptions of classes with which the algorithms for calculating ratings work are formed as follows: a set of objects $(S^1, \ldots, S^m)$ is chosen called learning objects or templates. On the basis of templates, descriptions of all objects S from S are formed in the form of matrices $D_{\text{ob}}(S) = \|\rho_t(S, S^k)\|_{m \times n}$, so that the description of an object is a set of "distances" to the templates, so that $I_{\text{ob}} = \mathsf{C}_{mn}(R_+)$. Note that similar designs in recent years are often used in the so-called "featureless" pattern recognition [4].

It is assumed that for learning objects there is information about their belonging to classes $K_1, \ldots, K_l$, that is, 0-1-vectors $(\alpha_{k1}, \ldots, \alpha_{kl})$, where the equality $\alpha_{kj} = 1$ is equivalent to the fact that the template S^k belongs to the class K_j. Vectors $(Al_{k1}, \ldots, Al_{kl})$ are called information vectors of learning objects. As descriptions of classes $K_1, \ldots, K_l$, data about belonging of the learning objects to them is used, i.e. $D_{\text{cl}}(K_j) = (\alpha_{1j}, \ldots, \alpha_{kj})$ for $j = 1, \ldots, l$ and hence, $\mathsf{I}_{\text{cl}} = \{0, 1\}^m$.

The elements of the information matrices $I = \{I_{ij}\}_{q \times l}$ are pairs $I_{ij} = (D_{\text{ob}}(S_i), D_{\text{cl}}(K_j))$, so the set of initial information I_i in this case is $C_{ql}(C_{mn}(R_+) \times \{0, 1\}^m)$.

$\mathsf{C}_{ql}(\{0,1\})$ are used as a space of final information I_f, and $\mathsf{C}_{ql}(R)$, as a space of estimates, so the recognizing operators for calculating estimates realize mappings of elements of the information matrix into real numbers.

In the heuristic family (model) of algorithms, the transition to 0-1-answers in the fundamental works [17–19] was carried out using one-parameter threshold decision rules, so that the main emphasis was placed on the model of recognizing operators, which we will denote M_Γ. This family of mappings from I_i to $C_{ql}(R_+)$ has the following parameters:

- family Ω of so-called support vectors or ω-projections, that is, subsets of the index set $\{1,\dots,n\}$, described by the set $\omega_1,\dots,\omega_N$ of o-1-vectors of length n such that $\bigcup_1^N \omega_r = \{1,\dots,n\}$,
- vector of "weights" of support sets $\vec{p} = (p_1,\dots p_N)$,
- vector of "weights" of objects of recognition $\vec{\gamma} = (\gamma_1,\dots,\gamma_m)$,
- vector of thresholds for metric discretization $\vec{\varepsilon} = (\varepsilon_1,\dots,\varepsilon_n)$,
- vector of "estimation strategy" $\vec{x} = (x_1, x_0) \in \{(1,0),(0,1),(1,1)\}$.

When calculating the estimates for given $\omega_n, \vec{\varepsilon}$, the so-called proximity functions on objects are used, defined in the classic version as follows:

$$b_{\omega_r,\vec{\varepsilon}}(S_1,S_2) = 1,$$

if for all t from $B\{1,\dots,n\}$ such that in $\omega_r = (\omega_r^1,\dots,\omega_r^n)$ the equality $\omega_r^t = 1$ is true together with the inequality $\rho_t(S_1,S_2) \le \varepsilon_t$, and

$$b_{\omega_r,\vec{\varepsilon}}(S_1,S_2) = 0,$$

otherwise.

In other words, the object S_2 is by definition close to S_1 if and only if it lies inside the $\vec{\varepsilon}$-"parallelepiped" with the center at S_1 in the ω-projection. Note that in numerous subsequent papers, other definitions of proximity functions were considered.

For the calculation of the final estimation Γ_{ij} of the object S_i in relation to the class K_j (that is, the element ij of the estimation matrix $\hat{I}_e$) first the auxiliary values Γ_j^1 and Γ_j^o for all $j = 1,\dots,l$ are calculated:

$$\Gamma_j^1 = \sum_{r=1}^{N} p_r \sum_{k=1}^{m} \alpha_{kj}\gamma_k B_{\omega_r,\vec{\varepsilon}}(S_i, S^k)$$

and

$$\Gamma_j^0 = \sum_{r=1}^{N} p_r \sum_{k=1}^{m} (1 - \alpha_{kj})\gamma_k (1 - B_{\omega_r,\vec{\varepsilon}}(S_i, S^k)).$$

Finally, the estimation Γ_{ij} is calculated:

$$\Gamma_{ij} = x_1 \Gamma_j^1 + x_0 \Gamma_j^0.$$

The family of estimation calculation algorithms combines important heuristic ideas of classification problem-solving. Reference sets or projections reflect the idea of autonomous use of individual fragments or aspects of the initial information. For example, in the problems of medical diagnostics, these may be combinations of features related to individual organs or body functions (syndromes), in the problems of processing and analyzing images, areas on a raster, etc. The weights of the support sets and templates reflect the hypothesis of their various "representativeness" or "informativeness", the summation of estimates comes from the idea of "voting". Finally, the use of "auxiliary" assessments $x_1\Gamma_j^1$ and $x_0\Gamma_j^0$ reflects the heuristic considerations of such kind: "for" the inclusion of an object S in the jth class "votes" the kth learning object from this class close to it, introducing "its vote" with weight γ_k into the sum Γ_j^1, or "for" the inclusion of an object S into the jth class "votes" not close to it kth learning object not from this class, introducing "its vote" of weight γ_k into the sum Γ_j^0 ("positive" use of "negative" examples).

To study the question of constructing correct algorithms for regular problems, in the original fundamental works [10, 21] the following simple statement was used: if within the framework of the F-extension of the model of recognizing operators $\mathsf{M}(\pi)$ for the regular task Z with the information matrix $\hat{I}$, we can build an arbitrary preassigned real matrix, i.e. if equality holds

$$\mathsf{F}(\mathsf{M}(\pi))(\hat{I}) = \mathsf{C}_{ql}(R),$$

and if C is a surjective mapping from $\mathsf{F}(\mathsf{M}(\pi))(\hat{I}) = \mathsf{C}_{ql}(R)$ on $C_{ql}(\{0,1\})$, then the problem Z is obviously solvable within the framework of the superposition family $C \circ \mathsf{F}(\mathsf{M}(\pi))(\hat{I})$, or, which is the same, there is a permissible for Z mapping f in $C \circ \mathsf{F}(\mathsf{M}(\pi))(\hat{I})$ such that $f(\hat{I}) = \hat{\hat{I}}$.

For a specific problem j with an information matrix $\hat{I}$, recognizing operators for calculating estimates from the (continual!) family M_Γ generate for Z, under the full variation of the parameters $\Omega, p, \gamma, \varepsilon$ and x, a set of real $q \cdot l$-matrices $M_{\Gamma(\hat{I})}$. As a family of corrective operations F, we used, among others, a set A_n of polynomials of degree at most n over real matrices with Hadamard multiplication. Regular problems were defined as problems whose information matrix satisfies the following "natural" condition: for any different j_1 and j_2 from $\{1, \ldots, l\}$, there is k in $\{1, \ldots, m\}$ such that $\alpha^1_{kj} \neq \alpha^2_{kj}$ and for any different i_1 and i_2 from $\{1, \ldots, q\}$ there is k in $\{1, \ldots, m\}$ and there is t in $\{1, \ldots, n\}$ such that $\rho_t(S_{i_1}, S^k) \neq \rho_t(S_{i_2}, S^k)$. For regular problems, the possibility of constructing an arbitrary matrix $\mathsf{C}_{ql}(R)$ in a polynomial extension of a bounded degree of a family M_Γ, and hence, the desired solvability of all regular problems was proved.

The possibility of the exact solution of all regular problems within the framework of this family of algorithms in the algebraic approach is defined as the property of its completeness discussed in more detail in the last section of this article.

25.4 On Bases and Problem-Oriented Bases

The concept of a base was first defined in the course of the development of an algebraic approach when considering the "last step" of constructing correct algorithms for the classification problems that are regular according to Zhuravlev. Considering that only mappings from $\mathsf{M}(\mathrm{I}_{\mathrm{str}})$ can be used, we immediately get that the necessary requirement for the existence of regular tasks is the requirement of equality

$$\mathsf{M}(\mathrm{I}^u_{\mathrm{str}})(\mathsf{I}_{e_n}) = \mathsf{I}_f.$$

In case when $\mathsf{M}(\mathrm{I}^u_{\mathrm{str}})$ are defined as sets of morphisms of specific categories [6], it leads to the definition of full categories Ψ_{ql}. In this case, for arbitrary not singleton sets U and V, the set $\mathsf{M}(I^u_{\mathrm{str}})$ is the unification of the sets $Hom_\Psi(\mathsf{C}_{ql}(\mathsf{U})^n \quad \mathsf{C}_{ql}(\mathsf{V})(\mathsf{C}_{ql}(\mathsf{U}))$ for all $n = 1, 2, \ldots$ and is denoted by $H(U, V)$, so it comes down to the condition of equality

$$H(U, V)(C_{ql}(U)) = C_{ql(V)}.$$

Note that the requirement of completeness usually turns out to be easily fulfilled as well as the requirement of "admissibility", which is formulated as the requirement that the introduced categories must be closed with respect to products and diagonalizations of morphisms, the latter are matching categorical conditions mappings between $q \cdot l$-matrices over arbitrary not singleton sets and all Cartesian degrees of such spaces.

Further, we note that at the "penultimate step" of constructing the desired algorithm within the framework $\mathsf{M}(I^u_{\text{str}})$, the possibility must be ensured to construct any element of the estimate space I_{e_n} (without loss of generality, we assume that $\mathsf{I}_{e_n} = \mathsf{I}_{e_f}$) by applying functions from $\mathsf{M}(I^u_{\text{str}})(\mathsf{I}_{e_{n-1}}, \mathsf{I}_{e_n})$ to the elements of the set $\mathsf{I}_{e_{n-1}}$ included in the "multi-image" of the initial information space.

This is what defines the concept of a base. In general: when constructing algorithms in the form of superpositions and applying them to the initial information in a specific problem, a series of transitions from one estimation space to the next estimation space is implemented, in each of which a corresponding "multi-image" is generated, i.e. a subset obtained as a set of images given by algorithmic operators and /or correcting operations. The requirement of sufficiency of those "multi-images" for the construction of an admissible solution as an element of the space of the final information determines the essence of the concept of a base.

In the initial version, the bases in the space of matrices $C_{ql(U)}$ were defined as the sets of matrices X, which, when mapped with the help of all the morphisms of complete admissible categories, were sufficient to generate all matrices from the original space. Moreover, it turned out that it is necessary and sufficient for generating all matrices from other spaces over arbitrary not singleton sets.

The concept of base used here corresponds to the concept of "a set of generators" in algebraic systems [1]. Note that the definition of a base does not include the requirement of a dead-end (irreducibility) and, all the more so, minimality of the set X. This, among other things, reflects the non-uniqueness of solutions inherent in the entire described research area, which leads to the possibility of using various kinds of regularization using additional heuristic functionals.

As a simple but important example, we describe the category Φ_0. Its objects $C_{ql}(U)$ are spaces of ql-matrices over arbitrary not singleton sets $\underline{U}$ and all Cartesian powers of such spaces. Sets of

morphisms are defined as sets of "elementwise" operations or functions. Formally, $Hom_{\varphi_0}(\mathsf{C}_{ql}(\mathsf{U})^m, \mathsf{C}_{ql}(\mathsf{V})^n)$ is defined as a set of mappings U from $\mathsf{C}_{ql}(\mathsf{U})^m$ into $\mathsf{C}_{ql}(\mathsf{V})^n$ such that there is a set of functions

$$f_k : (\mathsf{U})^m \to (\mathsf{V})^n, \quad k = 1, \ldots, n,$$

such that for all the sets of matrices

$$(\hat{U}^1, \ldots, \hat{U}^m) = (\|U_{ij}^1\|, \ldots, \|U_{ij}^m\|) \quad \text{from} \quad \mathsf{C}_{ql}(\mathsf{U})^m$$

the equation

$$(\hat{U}^1, \ldots, \hat{U}^m) = (V^1, \ldots, V^n) = (\|V_{ij}^1\|, \ldots, \|V_{ij}^n\|)$$

is equivalent to the fact that

$$\hat{V}_{ij}^k = f_k(U_{ij}^1, \ldots, U_{ij}^m).$$

for all $i = 1, \ldots, q$, $j = 1, \ldots, l$, and $k = 1, \ldots, n$.

Such, for example, are the operations of multiplying real matrices by a scalar, their addition, Hadamard multiplication, and multidimensional monotone splines.

For the category Φ_0, the set of matrices X is base if and only if for any different pairs of indices ("places in the matrix") $(i_1, j_1) \neq (i_2, j_2)$ in X there is a matrix $\hat{U} = \|U_{ij}\|$ such that $U_{i_1 j_1} \neq U_{i_2 j_2}$. The base X can also consist of a single matrix with pairwise different elements, but it can also contain a $(ql-1)$-matrix and at the same time be a dead-end one. From the point of view of the complexity of researching the procedures for constructing bases of the category Φ_0, we note that the corresponding proofs are reduced, for example, to constructing a matrix for $j_1 \neq j_2$, in which $U_{i_1 j_1} \neq U_{i_2 j_2}$ for all j_1 and j_2, and to the constructing, for i_1, i_2 and an arbitrary fixed j from $\{1, \ldots, l\}$, a matrix with $U_{i_1 j} \neq U_{i_2 j}$.

For some categories, situations are possible when a transition to a wider category does not lead to the emergence of new bases, i.e. does not lead to the expansion of sets of regular problems. In particular, this is the case when passing from the category Φ_0 to the wider symmetric category Σ_0, whose morphisms are mappings that commute with all permutations that "rearrange" the elements of ql-matrices. In addition to the "elementwise" morphisms of the category Φ_0 in Σ_0, the morphisms are, in some cases, the normalization operators of

real matrices or the unary Max operator used in [16], which selects the maximal element in a real matrix.

Earlier it was said that the solvability conditions for a problem Z with some information about precedents are described by the formula:

$$\mathsf{M}(\mathrm{I}^{\mathrm{pr}}_{\mathrm{str}})(\mathsf{I}_i, \mathsf{I}_f) \cap \mathsf{M}(I^u_{\mathrm{str}})(\mathsf{I}_i, \mathsf{I}_f) \neq \varnothing.$$

What is significant here is that the requirement of admissibility of the desired mapping, described by a set of precedents, makes sense only for functions from I_i into I_f, and the remaining requirements determine the corresponding sets of mappings $\mathsf{M}(I^u_{\mathrm{str}})(\mathsf{I}^{k-1}_e, \mathsf{I}^k_e)$, denoted as M^k_u, when forming a chain of superpositions:

$$\mathsf{Ii} \xrightarrow{\mathsf{M}^1_{\mathrm{u}}} \mathsf{I}_{e1} \xrightarrow{\mathsf{M}^2_{\mathrm{u}}} \mathsf{I}e_2 \longrightarrow \cdots \xrightarrow{\mathsf{M}^{\mathrm{n}}_{\mathrm{u}}} \mathsf{I}_{en} = \mathsf{I}_{\mathsf{f}}$$

The concept of a problem-oriented base arises due to the fact that, generally speaking, for the solvability of each specific problem, it is not required to ensure the possibility of constructing all elements of the space of final information I_f. Assuming that the domain $\pi(Z) \subseteq I_f$ is defined for the task Z, we can determine that the problem-oriented base X_k in the space I^k_ewill be a subset of this space such that it will be possible to build any element of $\pi(Z)$ within the reduced chain of superpositions.

For the case of problems in which the sets $\pi(Z)$ are related to elements of the initial information space I_i, called the problems with explicit set-theoretic constraints, a number of results were obtained in [8].

At the level of the conceptual framework, it should be noted once again that when constructing multilayer structures or, equivalently, chains of superpositions, it is of fundamental importance to study the corresponding chains of "multi-patterns" as subsets of virtual estimation spaces from the standpoint of their sufficiency for building a correct algorithm. It should be borne in mind that at some step the loss of the "sufficiency" being discussed turns out to be unrecoverable when moving "forward", which inevitably leads to the need to use various iterative methods.

25.5 On the Completeness of Algorithms and Families of Correcting Operations

When considering in the previous section the chains of transitions from the initial information space to the final information space through the estimation spaces, we mainly used "whole" sets of mappings that satisfy universal constraints. In the case of categories, these are the sets of type $H(U,V)$, defined above as the combinations by the arness n of all the sets of morphisms from $(C_{ql}(U))^n$ into $C_{ql(V)}$. These sets, convenient for theoretical application, naturally, cannot be used on their own in the synthesis of algorithms, which makes it necessary to study their (parametric) subsets. We have given the term "parametric" in parentheses, since for the time being we will be talking simply about subsets of "whole" sets of mappings.

The basic requirement for the subsets in question in the algebraic approach is called the completeness requirement. It is defined through the bases and is reduced either to the condition that the base's "multi-image" was again the base in the next space (weak completeness), or the "whole" space was the "multi-image" of the base. These properties of the sets of mappings $\mathsf{M} \subseteq H(U,V)$, provided that the subset X of the space $(C_{ql}(U))^n$ is a base, can be described as follows:

$$\mathsf{M}(X) = \{F(\hat{U}^1, \hat{U}^2, \ldots, \hat{U}^n) | F \subseteq M, (\hat{U}^1, \hat{U}^2, \ldots, \hat{U}^n) \subseteq X^n\}$$

is a base in $C_{ql(V)}$, or $M(X) = C_{ql(V)}$.

It is important that in the case of complete admissible categories, the following statement is also true: if the subset X of the space $(C_{ql}(U))^n$ is not a base, then $M(X)$ cannot be a base in $C_{ql(V)}$. At the formal level, this confirms the remark made at the end of the previous section.

At the substantive level, we note that the properties of weak completeness and completeness ensure that there are "enough" mappings in the studied families to ensure the existence of correct algorithms. From the point of view of solving regular problems, such families M are "not worse" than "whole sets of morphisms" of type $H(U,V)$ in

the sense that if a regular problem is solvable in the framework of $H(U, V)$, then it is also solvable within the framework M.

Obviously, the superfamilies of complete families are also complete, so that the "minimal" ones in some sense in terms of the composition of the subfamily of complete families are of interest. A number of results of this kind were obtained, for example, in [7].

In some cases, the completeness property that ensures the solvability of all regular problems is sufficient to guarantee the solvability within their framework of all problems from a wider set of solvable problems. It turns out, in particular, that this is true for models of the algorithms of the category Φ_0. What is interesting, this is generally not true for models of the category Σ_0 [6].

In the original papers on the algebraic approach [17–19] by using "direct" technically quite complex constructions, completeness was, in particular, proved for polynomial extensions of the abovementioned model of algorithms for calculating estimates and models of algorithms based on linear and piecewise linear separating surfaces in R^n. After the emergence of concepts of bases, using the fact of the necessary completeness of the family of superpositions of weakly complete models of algorithmic operators and complete families of correcting operations, in [6] and subsequent works, it was technically very easy to get a number of similar results for minimally complex families of maps.

25.6 Conclusion

This chapter describes the conceptual framework of the algebraic approach to the synthesis of correct algorithms using heuristic models. For multilayer algorithmic constructions, it is proposed to use the realizations of multi-images of the initial data on successive sections of the computational process (in successive estimation spaces) for the regular investigation of the problem of solvability preserving. These multi-images are used as a constraint when formulating successive optimization problems. It is also concluded that it is possible to use situations with the loss of solvability as the moments when a return to the previous virtual layers of the computational process is necessary.

References

[1] Mal'tsev, A.I. (1954) On the general theory of algebraic systems. *Matematicheskii Sbornik* (*N.S.*), 35(77), 1, 3–20.

[2] Mazurov, V.D. (1992) Models of contradictory data interpretation and committee method. *Trudy Instituta Matematiki i Mekhaniki UrO RAN*, 1, 193–203.

[3] Mazurov, V.D., Khachai, M.Y., and Rybin, A.I. (2002) Committee constructions for solving problems of selection, diagnostics, and prediction. *Trudy Instituta Matematiki i Mekhaniki UrO RAN*, 8(1), 66–102.

[4] Mottl, V.V, Dvoenko, S.D., Seredin, O.S., Kulikowski, C., and Muchnik, I. (2001) Featureless pattern recognition in an imaginary hilbert space and its application to protein fold classification. In Perner, P. (ed.) *Machine Learning and Data Mining in Pattern Recognition. MLDM 2001.* Lecture Notes in Computer Science, Vol. 2123 (Springer, Berlin).

[5] Rasstrigin, L.A. and Erenstein, L.A. (1981) *A Method of Collective Recognition* (Energoizdat, Moscow).

[6] Rudakov, K.V. (1988) Application of universal constraints in the analysis of classification algorithms. *Cybernetics and Systems Analysis*, 24(1), 1–6.

[7] Rudakov, K.V. (1986) Some universal restrictions for classification algorithms. *Zhurnal Vychislitel'noi Matematiki i Matematicheskoi Fiziki*, 26(11), 1719–1730.

[8] Rudakov, K.V. and Chekhovich, Y.V. (2003) Algebraic approach to the problem of learning algorithm synthesis for trend extraction. *Doklady Russian Akademii Nauk*, 388(1), 33–36.

[9] Rudakov, K.V. and Chekhovich, Y.V. (2005) Completeness criteria for classification problems with set-theoretic constraints. *Zhurnal Vychislitel'noi Matematiki i Matematicheskoi Fiziki*, 45(2), 344–353.

[10] Rudakov, K.V. and Vorontsov, K.V. (1999) On the optimization and monotone correction methods in the algebraic approach to the pattern recognition problem. *Doklady Russian Akademii Nauk*, 367(3), 314–317.

[11] Shannon, C.E. (1948) A mathematical theory of communication. *Bell System Technical Journal,* 27, 379–423, 623–656.

[12] Shannon, C.E. (1949) Communication in the presence of noise. *Proceedings of Institute of Radio Engineers*, 37(1), 10–21.

[13] Takhanov, R.S. (2010) A dichotomy theorem for the general minimum cost homomorphism problem. *27th International Symposium*

on Theoretical Aspects of Computer Science (*STACS 2010*), *Leibniz International Proceedings in Informatics* (*LIPIcs*), Vol. 5 (Schloss Dagstuhl–Leibniz–Zentrum fuer Informatik, Dagstuhl), pp. 657–668.

[14] Takhanov, R.S. (2010) Extensions of the minimum cost homomorphism problem. *16th Annual International Conference on Computing and Combinatorics* (*COCOON*). Lecture Notes in Computer Science (LNCS), Vol. 6196 (Springer), pp. 328–337.

[15] Zhuravlev, Y.I. (1977) Algebras over sets of incorrect (heuristic) algorithms. *Doklady Akademii Nauk SSSR*, 235(4), 761–763.

[16] Zhuravlev, Y.I. (1978) An algebraic approach to recognition and classification problems. *Problems of Cybernetics* (Nauka, Moscow), 33, 5–68; Hafner (1986).

[17] Zhuravlev, Y.I. (1977) Correct algebras over sets of incorrect (Heuristic) algorithms. I. *Cybernetics and Systems Analysis*, 13(4), 489–497.

[18] Zhuravlev, Y.I. (1977) Correct algebras over sets of incorrect (Heuristic) algorithms. II. *Cybernetics and Systems Analysis*, 13(6), 814–821.

[19] Zhuravlev, Y.I. (1978) Correct algebras over sets of incorrect (Heuristic) algorithms. III. *Cybernetics and Systems Analysis*, 14(2), 188–197.

[20] Zhuravlev, Y.I. and Nikiforov, V.V. (1971) Recognition algorithms based on computation of estimates. *Cybernetics and Systems Analysis*, 7(3), 387–400.

[21] Zhuravlev, Y.I. and Rudakov, K.V. (1987) On the algebraic correction of the information processing (transformation) procedures. *Problems of Applied Mathematics and Informatics* (Moscow, Nauka), pp. 187–198.

© 2025 World Scientific Publishing Company
https://doi.org/10.1142/9789811267215_0026

Chapter 26

Multilevel Models of Active Vision in Image Analysis

Mikhail V. Petrushan*, Mikhail V. Kopeliovich†, Konstantin B. Kalinin‡, Boris M. Vladimirskiy§, Dmitry G. Shaposhnikov¶, and Anatoly I. Samarin‖

Research Center of Neurotechnologies, Southern Federal University, Rostov-on-Don, the Russian Federation

**drn@bk.ru*

†*kop@km.ru*

‡*koster-mobile@mail.ru*

§*bvladimirski@gmail.com*

¶*dgshaposhnikov@sfedu.ru*

‖*samarin@krinc.ru*

The historical development of the physiology and psychology of perception and the technical area of "computer vision" periodically overlap, and both areas are mutually enriched by conceptual and model representations. In particular, retina-like sensors appeared in computer vision, which, as it turned out, not only model the features of the structural organization of the organs of vision but are also effective for the technical description and recognition of fragments of visual scenes. On the other hand, models created as a result of the generalization and systematization of physiological and psychological knowledge, receive technical validation are implemented as computer vision algorithms. One of the concepts — active perception, which appeared in psychology, was effectively used in technical vision. Within the framework of this concept, perception is treated not as a passive act, conditioned by reaction to events in the environment, but as an active behavior initiated by the

subject of perception, conditioned by internal motivation, and affected by the situational context. The use of such a concept in technical vision gives an effective, biologically-inspired solution to the issue of interpreting ambiguous sensory information when such ambiguity is reduced due to the actions of the observer. The implementation of the paradigm of active perception from a technical point of view leads to a certain structural organization of sensor models, as well as ways of processing and aggregating visual information. In this chapter, we consider models of active vision, models of visual sensors, algorithms for the lower and upper levels of visual information processing, as well as application of active vision models for various technical problems.

Keywords: Active vision, image analysis, descriptor, deep learning, bionic approach, visual uncertainty, objectivity and constancy of perception, navigation behavior, selectivity of observation, visual sensor, foveal sensor, inhomogeneous filter, behavioral vision model, feature maps, key points detection, criterion of description completeness, criterion of description uniqueness, viewing trajectories.

26.1 Introduction

The problems of image analysis and recognition arose and were successfully solved with the advent of technical means for recording images and computational means for their analysis. Since the 1950s, a statistical approach to image recognition has prevailed. Formally, the recognition procedure consists of the following set of actions. When analyzing the image, a set of its characteristics (primary features) is defined. In addition to the features obtained through image analysis, there is also a distinguishing feature, or a group of features, which are called classification features. Determining their values for a given set of primary features is the task that natural and artificial recognition systems perform. In order to establish the values of these characteristics, it is necessary to have information about how the known characteristics are related to the classification ones. Information about this relationship is given in the form of precedents — a set of descriptions (sets of features) of objects with known values of classification characteristics. Based on this prior information, it is also required to construct a decision rule that will assign a value of classifying characteristics to the object in accordance with its feature set. In the theory of pattern recognition, a set of features extracted by image analysis is called a descriptor. The algorithm that specifies the

decision rule and calculates the value of the classifying characteristics by a known descriptor is called the classifier. There is a wide variety of methods for describing an image (constructing descriptors) and classifying descriptors. For example, within the framework of solving one of the historically first tasks of pattern recognition — the classification of images of printed letters and digits — the descriptor was calculated as follows. A grid was drawn on a black and white image of the symbol, formed by the intersection of a number of regular vertical and horizontal lines. The presence of symbol pixels under the nodes of this grid was considered as a feature of the image. That is, if the first grid node is on the background pixel, then the first feature's value is zero; if this node is on the symbol's pixel, then the value of the first feature is equal to one. In the same way, the values of the remaining features are computed — as the presence of the symbol pixels under the remaining grid nodes. More complex features, often used as descriptor elements, characterize the color composition [34] or the form [3] of the object and describe the distribution of gradients [24] or other brightness heterogeneities [50] in the image. Most commonly used classifiers belong to one of four groups: metric-based, separating, probabilistic, and structurally logical. Metric-based classifiers evaluate the measure of the difference of the descriptor being analyzed from the descriptors of reference objects. The measure of difference is often formalized in the form of an Euclidean distance between descriptors in a feature space. The decision rule of the metric-based classifier assigns the analyzed object to a class whose descriptor of the reference object is closest (the minimal measure of difference) to the descriptor of the object under consideration. An example of a metric-based classifier is the k nearest neighbors algorithm. Separating classifiers utilize methods of dividing the feature space by hyperplanes or other hypersurfaces. Such hypersurfaces divide the feature space into domains of belonging to different classes. These hypersurfaces are determined as a result of a training procedure on a set of precedents. An example of a separating classifier is the perceptron.

Probabilistic classifiers, which include, for example, Bayesian ones, are based on the maximization of the so-called likelihood function describing the probabilities of the descriptor belonging to different classes. The structurally logical approach assumes the description of the object in the form of a set of attributes with spatial (or other) relationships between them. Often such a description is formalized in the form of a graph, where the features correspond to the nodes,

and the relationship between the features, to the edges. An example of such an approach is described in [10]. Within the framework of this chapter, we do not aim to describe in detail all the diversity of descriptors and classifiers. However, we consider it important to consider some historical trends in the development of methods for describing and recognizing visual images as they relate to the problem of active vision. As we highlighted above with the example of the symbol recognition problem, historically descriptors evolved from simple ones describing the presence of the pixels of the analyzed object at regular grid nodes to more complex ones that characterize the distribution of brightness inhomogeneities, color composition and the shape of the object in the image. In the 1990s and early 2000s, the opposite trend began to emerge, i.e., return to simple descriptors that characterize the lowest-level features. In the case of image recognition, these are the intensity values or the color components of the pixels. This opposite trend was caused by both a powerful theoretical background in the field of deep learning algorithms and the development of computer technology that allowed processing large volumes of data within acceptable times. Deep learning algorithms are able to construct complex features from simple ones; thus, it can be said that using this approach the descriptor is formed inside the classifier of deep learning. It is formed not according to the rules formalized by the trainer, but automatically within the training procedure. Deep learning networks, in particular, convolutional networks, have shown high efficiency in image recognition problems, but they require a large set of training samples (millions of precedents) and a lot of time for learning for some tasks, even on modern computers. Bionic approach, where the structural and functional characteristics of living organisms are embodied in technical devices, has shown high efficiency and promise as an alternative direction. At the level of the organization of perceptual systems, an example of a bionic approach would be spatially inhomogeneous filters, e.g. foveal filters whose structure imitates the spatial heterogeneity of the retina. The result of the aggregation of a complex of physiological and psychological representations was the concept of active vision, designed to describe the perceptual processes of living organisms and generalized for a class of technical systems. In contrast to the reactive concept of vision, in which the visual act is treated as a reaction to a certain stimulus, the concept of active vision postulates that the visual

act is purposeful, motivated, and therefore selective. Selectivity and purposefulness are not limited only to concentrating attention on some particular aspect of the visual scene, the current goal actually changes the process of perception itself, affecting all levels of processing of visual information, including the receptor level. Such a procedure is called pre-tuning in active vision. With respect to the recognition of images by technical systems, pre-tuning can consist of selecting the structure and parameters of the foveal filter and classifier. The selectivity of visual perception makes it possible to reduce the complexity of the pattern recognition problem, reducing it from a classification problem to a verification one. The system of active vision does not answer the question "what is this?", but determines whether the object in the field of view is the "wanted" one for the current visual task.

26.2 Models of Active Vision

Active vision approach, on the one hand, is the result of systematization of knowledge from the physiology and psychology of perception, and on the other hand, forms the basis of effective image recognition solutions that allow to circumvent certain limitations (for example, the need for a large training set) of other modern algorithms, such as deep learning classifiers. The paradigm of active vision appeared as a concept describing the process of human interaction with the environment, in which it is assumed that any, even purely sensory, visual function is represented in the information system of a person as a behavioral act. It is in line with axiomatic cognitive psychology [28] and ecological models of visual perception [12]. This approach was developed to describe human behavior in real-life environments; therefore, its application to the problems of analyzing three-dimensional scenes by systems of technical vision is reasonable. Compared to flat images, three-dimensional objects, and scenes have a number of principal features arising due to the interpretation ambiguity of their flat projections. This ambiguity is due not only to the fact that different objects can have the same visual projection from a particular angle but also to the effects of occlusion, shading, glares on objects, and distortion of their projections. Reducing such ambiguity in the interpretation of sensory signals can be achieved by organizing the active perception of three-dimensional scenes via viewing

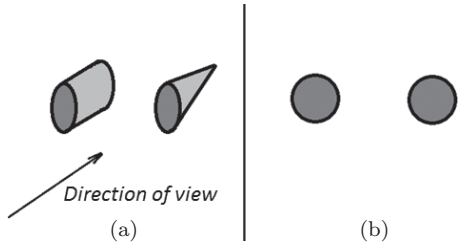


Figure 26.1. An example of visual uncertainty. Projections of different objects (b) look the same from certain point of view (a).

angle changes, movements in the environment, etc. For example, the removal of the uncertainty of visual perception shown in Figure 26.1 can be organized by changing the viewing angle to that at which the projections of the cylinder and the cone become distinct.

This example shows that motivation to initiate active visual behavior can be uncertainty in the interpretation of the observed visual image.

The implementation of active vision in technical systems allows reproducing such psychological phenomena as the objectivity and constancy of perception. The term "objectivity" refers to the property of the visual perception of a scene as a collection of separate objects. The property of vision, which allows the same (constant) perception of the scene objects during the observer's own motion subject to reticular projection changes caused by this motion, is called the constancy of perception. The phenomena of the objectivity and constancy of visual perception together allow a person or a technical system to perceive arbitrary three-dimensional scenes as a set of objects, regardless of the visual conditions and the type of the observer's movement. Within the active vision paradigm, object perception is interpreted as a behavioral phenomenon that takes place during the process of the observer's active behavior in the environment. Spatial localization of optical features is considered a criterion of object solidity, meaning that these features are compactly located in a three-dimensional space. Spatial localization is estimated using the method of constructing a map and a model of the environment [21], which performs translation of the localized features in the environment from the coordinate system of the robot (observer) into an external reference coordinate system. Thus, the problem of identity (belonging to one object) of different projections of the same object, which is

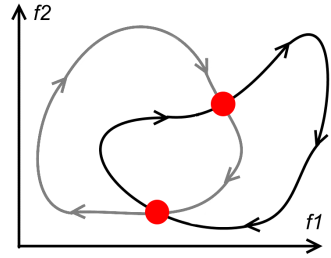


Figure 26.2. Possible descriptor trajectories in 2-dimensional feature space {f1, f2}. The first trajectory is in black, the second one, in gray. Circles denote the intersections of trajectories.

essentially generally unsolvable by methods of computer vision and pattern recognition, is easily resolved by the criterion of spatial localization of optical features observable from different directions of view.

"Constancy" is the perception of an object as the same even if its appearance changes due to changes in the viewing direction, illumination, partial occlusion, etc. When the location of the observer changes relative to the object, its appearance significantly changes, while the observer continues to perceive the object unchanged, i.e. constant. Such a property in technical vision systems is formalized as a forecast of the appearance of objects during observer's motion. The forecasting is performed using the methods of analysis of the trajectories of the object descriptor in the feature space when the observer moves relative to the object. Images of objects are described by some n-dimensional descriptor, which can be considered as a point in the n-dimensional feature space. When objects move relative to the visual system, the location of their descriptors in the feature space also changes. A simple case of the spatial motion of a two-dimensional descriptor {f1, f2} as a result of the motion of objects in the real world is presented in Figure 26.2. Displacements of the object in the real world relative to the visual system create a trajectory of the descriptor in the feature space. It is possible that some points on the trajectories of descriptors of different objects can be close to each other and even intersections of their trajectories are possible (Figure 26.2).

The proximity of trajectories or their intersection means that, under certain conditions, different objects can be indistinguishable when comparing their descriptors, since the descriptors are equal or very close to each other in the feature space. An illustration of such

a phenomenon is shown in Figure 26.1: when viewing a cylinder and a cone from a certain point of view, their images appear identical, which means that their descriptors are also the same.

One possible way to eliminate uncertainty is to perform an active observation procedure when the visual system moves relative to the object and the descriptors move along different trajectories in the feature space. Some patterns and properties of the descriptor trajectory can be used as a new high-level descriptor, which is used to enhance the recognition of an object in the process of active observation [18]. An object descriptor in this case is called a first-order descriptor, and the trajectory descriptor is a second-order descriptor. A similar approach was developed by Favorskaya [6]. She proposed to recognize the dynamic patterns by analyzing the trajectories of descriptors in the feature space.

Active vision models are also used to support navigation behavior of autonomous robots in real-world environments. Petrushan [32] has proposed a unified approach for solving problems of navigation and recognition of objects in a three-dimensional environment, based on a multilevel forecast and pre-tuning. The behavior of living organisms or robots in real-life environments is related to the prediction of sensory signals. Predicting sensory responses during the motion and analyzing the mismatch of predicted and real sensory data, one can distinguish between components of the sensory information related to the movement of the organism itself (or robot), and which are associated with changes in the environment. The forecast is modeled as a set of procedures of pre-tuning (or presetting) of the visual system.

Formally, the procedure for visual pre-tuning is associated with determining certain parameters of the recognition algorithms. The role of visual pre-tuning in recognizing two-dimensional images has been studied and described [42]. Such a procedure was interpreted as the expectation of the visual system to "see" a certain optical heterogeneity in a certain place. When recognizing a three-dimensional object, pre-tuning consists of expecting to "see" a certain appearance of the object when turning or moving the visual sensor. This type of pre-tuning is called active confirmation. The need to pre-tune the recognition algorithms also arises at lower levels of visual analysis. A general diagram linking the pre-tuning (presetting) and recognition procedures is shown in Figure 26.3.

According to Figure 26.3, the motion of a visual sensor generates a change in the projected image (the projection of the environment

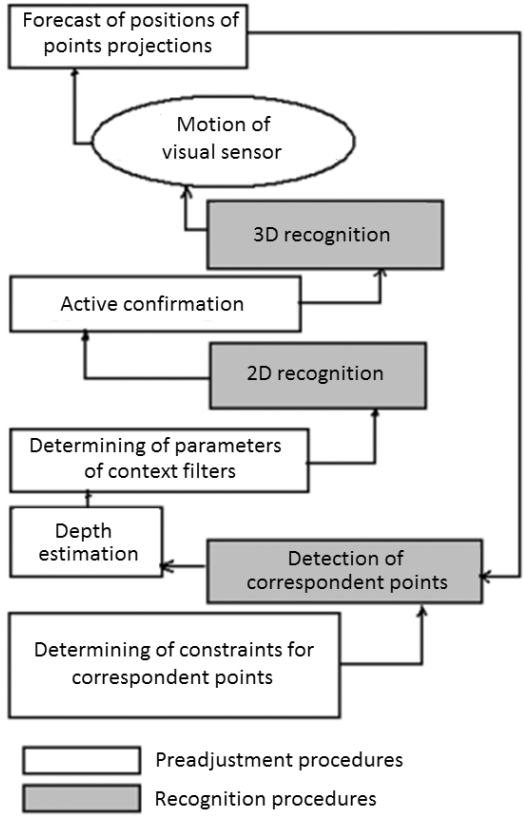


Figure 26.3. Inter-relationship between pre-tuning and recognition procedures.

onto the sensor's perception matrix). Knowing the nature of his movement, the observer can construct a model of motion of a projection image so as to determine in what directions its points will move. Next, the procedure for finding the matching points is performed. Matching points are such a pair of points in projected images (before the motion of the observer and after) that are projections of the same point of the environment. Deviations from their position from the predicted one will indicate the presence of motion of objects in the environment. Thus, by means of forecasting and comparison, projected image motion caused by the observer's own motion can be differentiated from changes caused by the movement of objects in the environment. The detection of matching points together with the known characteristics of the motion of the observer make it possible

to determine the distance to the objects in the environment [32]. After calculating the distance, certain parameters of context filters, which in this example are descriptor extractors, are tuned. This is done to provide invariance of descriptor to image scaling.

Summarizing the above examples of active vision application to various computer vision and robotics problems, one can note the following most important features of this approach:

(1) visual perception is not a separate act, but a stage in a complex behavioral cycle of the observer, closed through the environment (cf. example in Figure 26.3).
(2) Visual behavior is purposeful and selective — the observer "sees" and analyzes not the entire scene, but only what is necessary for the current task.
(3) Resolving the uncertainty of object recognition in an environment is achieved by active observation and forecasting of the appearance of object when the observer moves.
(4) Selectivity of the observation and the forecast are performed in the visual system in a form of the pre-tuning procedures, which formally consist of changing the parameters of the visual sensor (or descriptor extractor) and the classifier in accordance with the current task.

Furthermore, in the next sections of this chapter, various levels of visual behavior organization will be described in more detail, covering topics ranging from the visual sensor structure to the strategy of selecting areas of inspection.

26.3 Algorithms of the Lower Level of Processing in Active Vision Systems

26.3.1 *Types of sensors based on the spatially inhomogeneous representation of visual information*

One of the important directions of visual perception research is connected with the modeling of visual sensor, which structural and functional characteristics in many ways predetermine the process of perception. The most effective ones, both regarding the study of

visual perception properties and the applied use, are sensor models based on spatially inhomogeneous coding of visual information. Such sensors imitate one of the retina properties — a decrease in receptor cells density from the center to the periphery. Applying to image and video processing sensor models can be called *filters* or *operators of feature description* (*descriptor extractors*). Sensor spatial structure is modeled by so-called *input window* of a filter. The result of the processing of image fragment by a filter is called as *feature description* or *descriptor* of the fragment. In general, descriptor of a certain image area is a vector compiled from the values of a certain set of characteristics of the area. General scheme of image fragment processing by filter is shown in Figure 26.4. The example of filter input window is shown in Figure 26.4(a). Input window specifies areas for feature extraction. Applying filter to an image fragment one can extract feature values in each area specified by the input window. The vector of such features is the descriptor of the image fragment. Figure 26.4(c) illustrates an example of binary descriptor, indicating the presence of an image feature (e.g. intensity gradients) in sectors of the input window.

On the basis of sensors based on spatially inhomogeneous representation of visual information, a number of visual perception models allowing to reproduce realistic eye trajectories when viewing various images were developed. One of such sensors described in [42] is so-called foveal filter (see Figure 26.5), which structure contains central area modeling central retina area (fovea), and a number of peripheral areas. In each area, both central and peripheral, the histogram of orientations of intensity gradients in the image fragment, and respectively a set of histograms form feature vector of the image fragment. Foveal filter was used as a descriptor extractor for recognition of facial images. Foveal part of the input window was positioned on several key points in a facial image. Result of the processing of a facial image area in such key point by foveal filter was a descriptor of the area. The collection of such local descriptors was an integral descriptor of facial image. Comparison of integral descriptors was made by calculating weighted Euclidean distance between feature vectors.

Another example of spatially inhomogeneous filter is considered as part of behavioral vision model [40]. Spatial structure of its input window is shown in Figure 26.5(b). Histograms of intensity gradients are extracted from the intersections of radial and annular



Figure 26.4. A scheme of image fragment processing by feature description operator: (a) filter input window, (b) applying a filter to image fragment, (c) example of a possible descriptor.

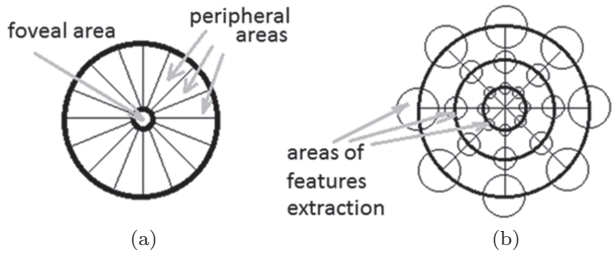


Figure 26.5. Different sensors models based on spatially inhomogeneous representation of visual information: (a) foveal filter [42], (b) model of retinal-like sensor of behavioral vision model [40].

lines ("areas of features extraction" in Figure 26.5(b)). In this case, the image, in which the filter input window is applied, undergoes a Gaussian blur with kernel, which size increases from the center to the periphery, thus modeling the spatial resolution of the retina. It was shown that behavioral vision model that includes formalization of the described spatially inhomogeneous sensor, allows synthesizing different types of trajectories of visual sensor movements corresponding to real types of trajectories of eye movements [35].

In addition to the active use of spatially inhomogeneous filters as models of visual sensors for the study of visual perception, such filters are used for applied problems of image analysis and recognition. One of the first spatially inhomogeneous descriptors which is effectively used for recognition of wide class of contour images is "shape context" descriptor [3], which spatial structure of input window is shown in Figure 26.6. When the input window of the "shape context" filter is applied to a fragment of the image (Figures 26.6(a) and 26.6(b)),

the number of contour points that fall in the area of the filter input window formed by the intersection of the radial and annular lines is calculated. The number of contour points is limited so that the contour consists of a certain number of points, e.g. one hundred. Example of possible descriptor is shown in Figure 26.6(c). Descriptor is a histogram of contour points distribution within areas of the filter input window. Comparison of the descriptors is performed by calculating of Euclidean distance between feature vectors. To compare different shapes Belongie *et al.* [3] also estimated power of transformation from one contour to another, apart from calculation of Euclidean metrics. The correspondence between contour points was estimated by means of comparison of all descriptors of points from the first contour to descriptors of points from the second contour by searching of minimum of Euclidean distance between descriptors. After that, the power of transformation was estimated, which lead to displacement of all points of the first contour to positions of the corresponding points of the second contour.

On the example of "shape context" descriptor, let's consider three general questions about the descriptions of image fragments by any descriptors:

Which image points (or fragments) should be described?

What size of the filter input window should be used?

How filter input window should be oriented (in the image plane)?

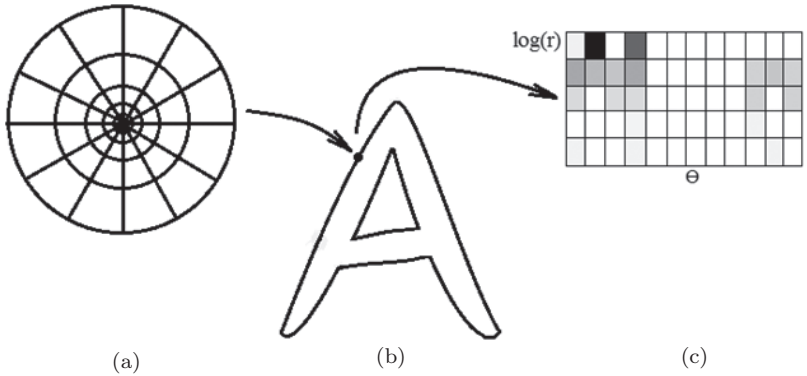


Figure 26.6. "Shape context" descriptor: (a) descriptor input window, (b) example of contour image and point of input window center, (c) visual representation of feature vector as the histogram of presence of shape contour points within the areas of the filter input window, which are formed by intersections of radial and annular lines.

Thus, common questions for describing an image with any filters are to determine the positioning points, scale, and orientation of the filter input window.

Regarding the shape context, the authors of this descriptor suggest to describe all points of the thinned (with limited number of contour points) contour. The orientation of the input window is determined by the direction from the described contour point to the next contour point. In this case, clockwise direction of contour bypass is selected. The size of the filter input window is determined by the size of the described shape since in the proposed recognition method it is assumed that there is only one object on the image, and therefore one contour, and this means that the size of the described object are known.

One of the most stable algorithms for determining the scale of a feature description was proposed by David Lowe [24]. To determine scale of the description, a number of image processing procedures are proposed (Figure 26.7).

First, a number of blurred images are created, representing the results of the convolution of the original image with a number of Gaussian kernels of increasing size. Then, the differences between neighboring Gaussian blur maps are calculated. Thus, a number of difference images are formed (in the author's work, called difference-of-Gaussian images). Next, in the array of difference images, the search for local minima and maxima is performed (Figure 26.8). The minimum or maximum in the current pixel (shown by the cross in Figure 26.8) is determined by comparing the values of the difference

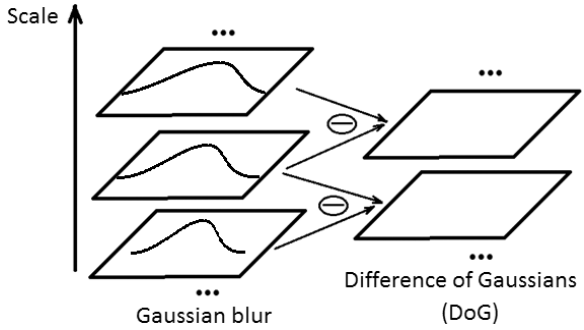


Figure 26.7. The sequence of image frames processing in SIFT. Explanations are given in the text.

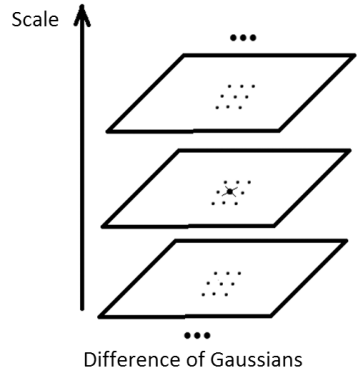


Figure 26.8. Selection of local extrema in maps of Gaussians difference. Explanations are given in the text.

images in this pixel and 26 neighboring pixels — 8 neighboring pixels in the current difference image and 9 neighboring ones in the difference images of smaller and larger scale. Thus, in addition to determining the key points for the description, the scale of the description is also determined.

The next considered filter, which the input window has spatially inhomogeneous structure simulating the density distribution of receptors on the retina is DAISY [46]. Structure of DAISY input window is shown in Figure 26.9. In each filter area, indicated by the bold circle, the histogram of intensity gradients is calculated. In this case, the orientation maps, calculated by convolution of the original image with the kernel of the gradient operator rotated by a certain angle, are blurred by convolution with a Gaussian kernel of different sizes. The size of Gaussian kernels increases from the center to the periphery.

In contrast to the DAISY, the spatial structure of the input window of BRISK descriptor [22] consists of disjoint areas (Figure 26.10), in each of which the image is blurred by convolution with a Gaussian kernel. The sizes of the Gaussian kernels increase from the center to the periphery to enhance the degree of blurring of the image at the edges of the input window.

Feature description generated by the BRISK is a set of binary features that characterize the sign of the intensity gradient between pairs of areas of the input filter window. If the gradient is positive then the feature sign is 1, and 0 otherwise.

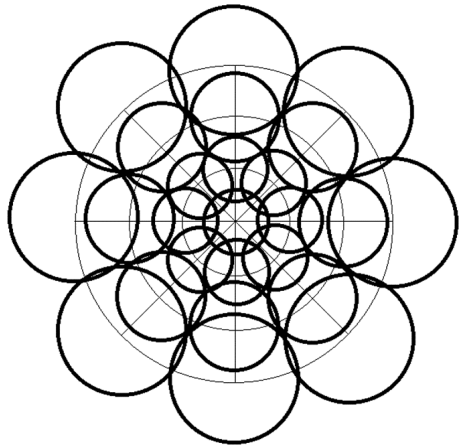


Figure 26.9. Structure of the DAISY input window. Explanations are given in the text.

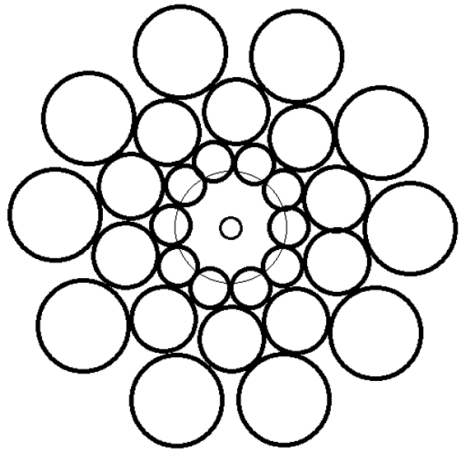


Figure 26.10. Approximate spatial structure of the BRISK input window. Explanations are given in the text.

A similar structure of the input window and a method of forming a feature vector as a set of binary features characterizing the sign of the intensity gradient between pairs of input window areas are typical for the FREAK descriptor [1]. 43 areas of the filter input window of the FREAK are arranged in three groups of rings with partial overlap, as in the DAISY (Figure 26.9).

26.3.2 *Construction of feature maps in the visual sensor structure*

One of the very fundamental and common objectives in a class of computer vision problems is related to feature space determination for making image descriptions and comparing them. Depending on a task, different feature spaces are used [8, 27, 52], matching the following criteria:

- Robustness, e.g. invariant to a number of object (image part) transformations, or its point of view, or in other words, invariant to in-class variations of object visual representation;
- Uniqueness, e.g. descriptors of different objects should differ from each other.

Feature descriptors, based on monochromatic intensity gradients, for example, ones that utilize gradient orientation histogram (HOG) at the point of interest, are widely used. More sophisticated algorithms for making image descriptions and comparing them, such as SIFT [24], also use the orientation of intensity gradient as first-level descriptors. Color information is also used in several descriptors. The effectiveness of such descriptors was shown on a problem of image classification by category [11, 19]. Obviously, color information gives an additional description of object being observed, that could be used to improve correct classification probability. Mostly, RGB components (red, green, blue) or HSV (hue, saturation, value) are used as color description of object.

One of the common methods of feature space construction involves using of Gaussian operator in the image or its gradient map to achieve smoothing effect. This leads to the suppression of local high-frequency image characteristics information, leaving mostly low-frequency components that describe the high-scale features of an object. Referencing to filters mentioned in Section 26.3.1, it shows that Gaussian kernel size differs from the filter window center to its peripherals. The spatial variability of blur when performing image convolution using different Gaussian kernels allows to describe the part of image at a filter center more detailed, leaving only generalized features in a peripheral part of filter window. This simulates foveal structure of retina.

Optimization of description in terms of uniqueness and robustness maximization is achieved not only by altering of structure and

parameters of filter input window but also varying the feature space being used. One of the ways to synthesize optimal feature space is to create the feature map as weighted sum of HSV components of color information: $M = k_1H + k_2S + k_3V$ [34]. The weight coefficients of sum k_1, k_2, k_3 come up within the optimization procedure, defined, for example, to minimize classification error rate on the test image base.

26.4 Algorithms of the Upper Level of Processing in Active Vision Systems

26.4.1 *Regions of interest*

Particular attention at investigation of biological and computer vision is given to problem of most informative image areas localization, because the quality of classification depends on the understanding how the most informative areas are detected.

A variety of methods for image key points detection operate with local features of a point surrounding area to determine how much the point differs from other points in the image. Several algorithms developed to detect corner points, which are one of the most stably detected ones within affine and perspective image transformations — Harris detector [13], Shi-Tomasi detector [47]. Corner points are the points of lines intersection at an angle close to 90°.

In addition to methods of detecting areas of interest (in the form of key points) based on the local image features (such as angles) detection, there are approaches based on the searching for informative areas depending on the specifics of the visual task [4]. Consider the problem of localizing the most informative areas on images that are optimal for the input window positioning of a foveal filter. The points of the input window center positioning are called fixation points referring to natural vision. We define two criteria for optimality. The first criterion is defined in terms of the completeness of the description. The ratio of information extracted when viewing the image to the complete information contained in this image depends on the coordinates of the fixation points. The criterion of the completeness of the description refers to the optimal viewing trajectory, which is characterized by the maximization of information extraction with simultaneous minimization of the number of fixation points. This criterion is formalized as a measure of the similarity between the

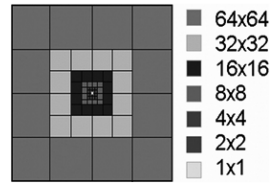


Figure 26.11. The input window of the foveal filter of intensity averaging. In each area of the window, intensity of the image fragment located inside of the area is averaged.

original image and the image restored from descriptors — the maximum similarity corresponds to the optimal distribution of fixation points. The second criterion is formulated in terms of the uniqueness of the description. This criterion is formalized as the total variability of the descriptor elements of the corresponding points for different objects within the same class.

The corresponding tasks from computer vision can be formulated as follows: a complete description of the image, when the description of the object image should be sufficient to reconstruct the image (from descriptors); and a unique description of the images, where the description of the object image should be distinguishable from descriptions of other objects within the same class.

To investigate the influence of the criterion of description completeness to the strategy of selecting fixation points, a simple foveal filter with a growing size of the intensity averaging fields from the center to the periphery is presented (Figure 26.11). Consider the task of determining the optimal viewing strategy to maximize the completeness of the description (in terms of the similarity measure of an image reconstructed from descriptors readings with the original one) while minimizing the number of fixation points on the image.

In each area of the filter input window (except the central part), the average intensity of the image fragment lying in this region is calculated. The central part of the input window contains 4×4 averaging fields, the size of each is 1×1 pixels. Each peripheral field covers a region of $2^N \times 2^N$ pixels, where N is 1–6. The twelve external peripheral fields cover areas of 64×64 pixels. The total number of averaging areas in such a spatially inhomogeneous filter is 88, covering a totally of 256×256 pixels of the image at each fixation point (the point of positioning of the filter input window center). The algorithm for image reconstruction from the descriptors is based on

finding the closest fixation point for each pixel of the restored image and assigning the intensity value of this pixel to the intensity value in the corresponding averaging area of the foveal filter input window (which covers this pixel) positioned at the nearest fixation point. The difference measure between the original and reconstructed images is determined by the formula:

$$D(X^I, X^R) = \sum_{i,j=1}^{i_m,j_m} |x^I_{i,j} - x^R_{i,j}|, \tag{26.1}$$

where X^I, X^R are original and restored images, respectively; $X^I_{i,j}$, $X^R_{i,j}$ are intensity at pixel coordinates (i, j) on the original and restored images, respectively; i_m, j_m are width and height of the original and restored image, respectively (the sizes of both images are assumed to be equal).

The three models of fixation point distribution of the foveal filter were compared based on 10 face images: uniform, random, and gradient. The gradient distribution of fixation points is defined by the two rules: the fixation points should be located in the region of a high diversity of orientations of local intensity gradients and the distance between any two fixation points should be greater than a given threshold in order to prevent the fixation points grouping in a single region having a high variety of gradient orientations. Gradients of intensity were calculated using the differentiation operator (Sobel). The number of fixation points tried was 25–49. The gradient distribution of the positioning points of the input filter window results in a more efficient description of the image in the terms of minimizing the difference measure (26.1) between the original and reconstructed image than the uniform or random distribution. In particular, the gradient distribution causes a high density of fixation points in the vicinity of the eyes, nose, and lips. The second criterion for choosing the optimal key points (the most informative areas), namely the criterion of the uniqueness of the description, is formalized as the cumulative variation (in a series of images of an object) of intensity in a set of fixation points $\{P\}$:

$$U = \sum_{i=1}^{N_{\text{fp}}} \nu(P_i), \tag{26.2}$$

where N_{fp} is the number of fixation points, $v(P_i)$ is the value of the intensity variation at the point P_i, calculated by the formula:

$$\nu(P_i) = \frac{1}{N_{\text{im}}} \sum_{j=1}^{N_{\text{im}}} |b(P_{i,j}) - M(b(P_i))|, \tag{26.3}$$

where N_{im} is the number of images of the object, $b(P_{i,j})$ is the pixel intensity value at point P_i on jth image, and $M(b(P_i))$ is the average (by the image selection) of the pixel intensity at point P_i.

The effectiveness of distribution of fixation points in terms of the description uniqueness maximizing can be estimated using this criterion. Only delocalized distribution of fixation points is considered (to prevent grouping of fixation points in a single region of high variability) such that the distance between each pair of points (Pi, Pj) is greater than the given threshold k:

$$\rho(P_i, P_j) \geq k, \quad \forall i \neq j, \tag{26.4}$$

where $\rho(P_i, P_j)$ is the distance between the points P_i и P_j of the image, which must be no less than a given threshold k.

The criterion of uniqueness of the description can be formulated as a question: which fixation points on the object image do cover the common features of objects of a given class, and which do cover individual features of an object that distinguish it from other objects of this class?

The task of selecting fixation points on the basis of this criterion is considered on the example of images of individuals using the method of constructing a generalized portrait [17]. The method of the generalized portrait is averaging the intensity in the same places of images of persons in a coordinate system tied to the points of the eye centers. Before averaging, each image is centered and scaled in such a way that the eye centers of each image are located in the same pixel coordinates. The intensity variation (26.4) was calculated at each point of the generalized portrait in order to determine the general and individual characteristics of the face images. Generalized portraits and maps of intra-class variation were obtained on a base of faces ($n = 30$) of men and women aged 18–25 years. Areas with high values of intra-class variation of intensity are located around the eyes, eyebrows, nose, and lips. The distribution of these areas obtained in accordance with the maximization of the description uniqueness

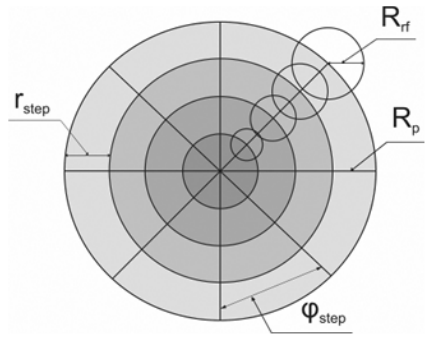


Figure 26.12. The scheme of input window. $\mathbf{R}_p$ is the size of the input window; $\mathbf{R}_{rf}$ — the size of context area of each node of the input window; $\mathbf{r}_{\text{step}}$ and φ_{step} are parameters of the sector and radial structures of input window.

criterion U is qualitatively similar to the gradient distribution of fixation points corresponding to minimizing the description completeness criterion D.

26.4.2 *Viewing trajectories*

It is well-known that the mechanisms of visual attention include many components (from sensory to cognitive) that interact with each other [2, 54, 55]. However, despite intensive research in this field in the last decade, carried out including modern methods of eye movement recording, the question of the basic mechanisms of attention and their hierarchy is not currently solved. Taking into account the inconsistency of known facts and hypotheses about the mechanisms for translating the view, mathematical models that simulate neurobiological phenomena are considered as one of the most important tools [29, 30, 35, 41, 53] in studying the mechanisms of visual attention.

We have developed a research model for the formation of viewing trajectories of complex images. The model includes an input window, a module of feature extraction, and an attraction function that determines the next fixation point. The input window is a structure that consists of n nodes located at the intersection of concentric circles and radial lines (Figure 26.12).

Each node is described by a set of features detected in its vicinity, the size of which increases from the center to the periphery, emulating

the resolution change in the human visual field. The detailed function of attraction (26.5) includes components that formalize the set of factors influencing the translation of the viewpoint related to image properties, field structure of vision, and cognitive processes (pre-tuning, previous experience, cognitive strategies, etc.).

$$F(x,y,h) = \sum_{i=1}^{n} k_i f_i(x,y) + V(Q, r_F, r_P)(x,y) + C \cdot \mathrm{Fix}(x,y,h), \tag{26.5}$$

where $f_i(x,y) - n$ feature maps that are results of image processing using standard methods (two-dimensional Fourier transform, calculation of local intensity gradients, etc.);

k_i — normalization coefficients for each feature map;

$V(Q, r_F, r_P)$ — an integral description of the field of view structure, determined from the data of psychophysical tests for a particular subject and including several components;

$Q(x,y)$ — a map of the structure of the field of view (sensory sharpness adjustment, local spatial heterogeneity [44], functional and morphological scotoma, etc.);

$r_F(x,y)$ — size (shape) of the central part of the field of view;

$r_P(x,y)$ — size (shape) of the peripheral part of the field of view;

C — coefficient depending on the nature of the current visual task (free observation, search for specified fragments, etc.);

$Fix(x,y,h)$ — a component that is dynamically formed from the results of processing fragments with previous fixations of the input window and can be described as the set of coordinates of consecutive fixations or as their weighted sum.

The choice of fixation points and the formation of a model viewing trajectory were carried out as follows. Each pixel of the image was assigned a value k:

$$k = \begin{cases} k_f, & \text{if } f(x,y) \neq 0 \\ 1, & \text{if } f(x,y) = 0, \end{cases} \tag{26.6}$$

where k_f is the coefficient of feature attraction, the value of which determines the significance of the analyzed local characteristic. In addition, if the current point also belonged to the most informative image area, the value of k was increased by k_{pr}. The choice of informative zones was carried out in accordance with the experimental data of A.L. Yarbus, as well as the results of their own tests.

The coordinates of the initial positioning point of the input window X_0 and Y_0 were set according to the experimental data. The values of k of all the pixels that got into the context region of each ith node were summed up, and the sum was normalized by the area of the given region. Resulting value is P_i. After that, the center of the input window was moved to the node with the maximum value P_i, and the values k for all pixels that are covered by the central (foveal) region of the input window, were decreased by 1.

The parameters of the proposed simulation model associated with the structure of the input window (the size of the foveal and peripheral visual field, the location of the scotoma) were determined by the results of psychophysical tests on a particular subject. The remaining parameters (the coefficients k_{pr}, k_f, C, the weights of the field of view, etc.) were optimized to reproduce the experimental results in order to determine the factor that leads to translation of the subject's viewpoint at each point of the viewing trajectory.

In psychophysical tests, three types of viewing strategies were found [36], which can be characterized by three kinds of trajectories — scanning, object-return and object-sequential. The computational experiments presented in this work were aimed at studying the parameters of the model to determine the possible factors for the formation of different viewing strategies. As a set of features and their combinations, we used the orientated intensity differences and the parameters of the CIECAM 97 color perception model. Simulation experiments were carried out using textured images ($n = 8$).

In the first series of experiments, the effect of the radial structure of the input window was investigated (see φ_{step} at Figure 26.12) having other parameters fixed. The angle of the sectors was 72° and 22,5°. The analysis of the obtained model trajectories ($n = 80$) showed that in most cases (up to 90%) when the radial structure of the input window is varied, the nature of the "observation" does not change. At the same time, having different angular structure, the direction of "observation" changed (Figure 26.13).

Thus, it can be assumed that the structure of the field of view does not have a big influence on the viewing strategy, but it can determine the preferred consequent direction of the sight.

In the second series of simulation experiments, the influence of local features of the image on the construction of trajectories was

evaluated. In this case, the coefficient of feature attraction k_f (see equation (26.6)) varied from 6 to 36. In this case, the structure of the input window and other parameters were constant.

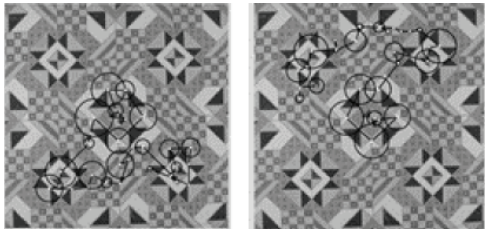
For each of the 8 test images, 11 model "observation" trajectories were constructed. In all cases, there was a clear tendency to change the nature of the trajectories, namely, when increasing k_f, the trajectory changed from the scanning type to the objective one (Figure 26.14).

The results of these experiments showed that an important factor determining the structure of the trajectory of viewing complex images is their indicative content. Namely, the higher the local feature value, the more it affects the attraction of the sight.

26.5 Applications of Active Vision Models in Various Technical Areas

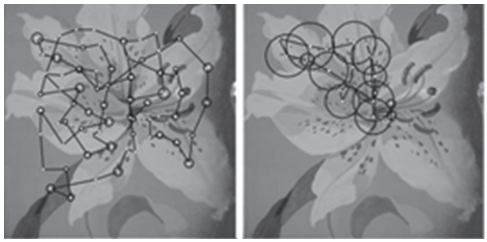
26.5.1 *Recognition of facial video images using the analysis of the trajectories of the descriptor*

Within active vision models, observation is the formation of a description of the object in the process of the observer's movement relative to the object of viewing. Such a motion generates a change in the object descriptor (first-order descriptor) in the feature space. The nature of such a change is described by a second-order descriptor and is used for additional object verification along with first-order



(a) (b)

Figure 26.13. An example of the influence of the input window structure on viewing trajectories. The angle of the input window sector was (a) 72° and (b) 22.5°.



(a) (b)

Figure 26.14. Artificial viewing trajectories for different values of the coefficient of attraction: (a) 10, (b) 35.

descriptors. Kopeliovich *et al.* [18] used this approach to recognize facial video images. The two first-order descriptors are considered: A simple descriptor (SD) and a Gradient descriptor (GD). The SD is the vector of the pixel intensity values of the image in the local vicinity of the point being described. Namely, the intensity of all the pixels in the rows of the rectangular region $R \times R$ around the central pixel of the face area on the image is expanded into a single row of n elements ($n = R^2$). The value of R was chosen equal to the width of the detected face in the image. To reduce computational complexity and make it possible to visualize the trajectory, the n-dimensional descriptor is projected onto the 2-dimensional feature space using the principal component analysis (PCA) method. Example of a simple descriptor trajectory computed on the VidTIMIT dataset [43] in the space of the first two principal components is shown in Figure 26.15.

Another first-order descriptor is the Gradient Descriptor [33]. It describes the local spatial distribution of the orientations of over-threshold gradients. Namely, gradients of intensity Grad_x, Grad_y (in horizontal and vertical directions respectively) are calculated at a given point using the Sobel operator. Next, the value (Magn) and the orientation (Orient) of the gradient are calculated as follows:

$$\mathrm{Magn} = \sqrt{\mathrm{Grad}_x^2 + \mathrm{Grad}_y^2}, \tag{26.7}$$

$$\mathrm{Orient} = \arctan\left(\frac{\mathrm{Grad}_y}{\mathrm{Grad}_x}\right). \tag{26.8}$$

The orientation angle is quantized into 16 sectors. Then the descriptor values $D(i, j)$, $i, j = 1 \ldots 16$ can be calculated as the number of gradients in the ith sector (total 16 sectors) of the descriptor

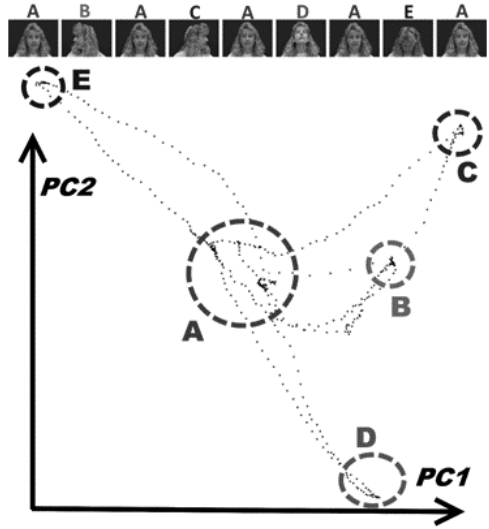


Figure 26.15. Examples of frames obtained from one video session of VidTIMIT dataset, and trajectories of Simple descriptor obtained for the session (in space of two principal components PC1 and PC2). Black dots correspond to values of projections of first-order descriptor. Head positions and corresponding areas in feature space are indicated as A–E.

window (Figure 26.16) so that their quantized orientations are equal to j, and the value is greater than the selected threshold T. In this work, the threshold was chosen as 10% of the maximum value of the gradient in the entire image. The center of the descriptor window is placed in the center of the face on the image, with a diameter of the descriptor equal to the width of the face in the image. The face region can be found by the Viola–Jones method [50].

Further processing involves projecting the descriptor into 2 principal components in the same way as it was done for the SD.

The three second-order descriptors (descriptors of trajectories) were investigated: Histogram of angles (AH), Histogram of transitions (TH), and Probability map (PM).

The AH is defined as follows. The relative movement of the observation system and the object leads to changing of object descriptor reading and its projection onto the principal components. The change in the projection of the descriptor forms a displacement vector that is directed at a certain angle (alpha and beta in Figure 26.17) to the



Figure 26.16. An example of input window positioning of Gradient descriptor.

first principal component (the X axis in Figure 26.17), e.g. — to the eigenvector of the descriptor covariance matrix, which corresponds to the highest eigenvalue. The histogram of such angles distribution is the first trajectory descriptor considered. All possible angles from 0° to 360° are quantized by 100. The element of index i of the histogram is the amount of the angles belonging to the ith quantum, relative to the angles computed along the whole trajectory.

The second trajectory descriptor considered is the Histogram of Transitions, which is defined as a 360×360 matrix. The value of the matrix element $\mathrm{TH}_{\alpha,\beta}$ is the fraction of the angles α, followed by the angles β in the entire descriptor path (see Figure 26.17). α, β are angles between consequent trajectory displacement vectors and the first principal component, rounded to integer values. After processing all adjacent segments of the path, the TH matrix is smoothed by the Gaussian filter of window 51 (the standard deviation is 8.5). Next, all elements of the matrix are transformed according to the following scheme: the minimal value found at the whole TH is subtracted from all its elements; elements are divided by the difference between the values of the maximum and minimum elements of the TH matrix. Such a transformation maps the source values of the elements to the interval $[0, 1]$. The resulting matrix is treated as a path descriptor.

The third variant of the trajectory descriptor is the Probability map (PM) of the transition from a certain region in the feature space

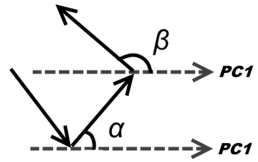


Figure 26.17. An example of sequence of vectors of descriptor trajectory in feature space. PC1-axis corresponds to the direction of the first principal component.

at a certain angle. It is defined as follows. For the observed object, the region in the 2-dimensional feature space, which includes all possible object descriptor projections onto the 2 principal components, is separated by a regular grid of 100×100. Each 3 consecutive points of the descriptor trajectory in the feature space form an angle that is quantized by 8 (0°–360° with 45° increment). The descriptor element $\mathrm{PM}_{i,j,\alpha}$ is the amount of the quantized transition of angles α among the entire trajectory located in the $[i, j]$ cell of the regular grid. For each angle α, part of the probability map $\mathrm{PM}_{i,j}$ corresponding to the angle α, is smoothed by the Gaussian filter of window 11 (the standard deviation is 1.8).

The difference between the trajectories of the descriptor is defined as the Euclidean distance between second-order descriptors (trajectory descriptors). The classification rule that determines whether the descriptor paths belong to the same class or to some other class is based on comparing their difference with the threshold value: if the distance is less than the threshold value, they are considered objects of the same class, otherwise they are treated as objects of different classes.

The efficiency of the methods of integral description of the trajectory of the descriptor considered was estimated within the problem of identifying faces on video recordings. The experimental base consists of 15 video records 20 seconds 30 fps each, recorded in grayscale using Logitech C920 webcam. Each frame of the video contains the head of the subject, who sits frontal to the camera and performs head tilt up-down. 5 subjects (men aged 24–36) took part in the experiment, and 3 video recordings were made per each. The background on the video recordings is constant and the same for the entire base; the illumination was about 500 lux. The distance from the web camera to the subject head was about 0.3 m.

To assess the quality of integral descriptors, the face area on each video was described by each of the descriptors considered. Then,

all descriptors were classified as one of the five classes (the number of subjects) by estimating the Euclidean distance to the reference descriptors. As the reference descriptors the descriptor values averaged within the same class were used. Varying the classification threshold, the so-called ROC curve (receiver operating characteristic, also often referred to as error curve) was obtained. The area under the ROC curve (AUC ROC) is a measure of the classifier quality. For the three integrated path descriptors, the following AUC ROC values were obtained on the test base described:

- Histogram of the angles of transitions — 0.96.
- Probability map for the transition between the angles — 0.91.
- Probability map for the transition from the area of the characteristic space at a certain angle — 0.96.

The defined approach can be used as an independent method of describing and recognizing dynamic images, and as a component in a cascade classifier to enhance methods of recognizing static images in the presence of motion of the objects under consideration.

26.5.2 *The use of models of generation of viewing trajectories for image recognition*

For the analysis of medical images, a behavioral model for the formation of viewing trajectories was developed (see Section 26.4.2). The viewing trajectories obtained during the computational experiments were compared with each other, after which, based on a quantitative assessment of the differences, it was concluded that the images belong to the same or another class.

When analyzing fragments of images the image and the color sensitivity perceived by the human eye were reduced from the fovea to the periphery at each model input window point of fixation (Figure 26.18). The level of blurring and the degree of color perception were determined according to Gaussian distributions with standard deviations σ_{blur} and σ_{color}, respectively.

The blur radius increases from the center to the periphery:

$$R_{\text{blur}} = k \cdot (1 - f(r, \sigma_{\text{blur}}))/f(0, \sigma_{\text{blur}}), \tag{26.9}$$

where r is the distance from the point to the center of the input window, k — proportionality coefficient, $f(r, \sigma_{\text{blur}})$ — Gaussian function.

Saturation, if the image is colored, decreases from the center to the periphery:

$$C = C_0 \cdot f(r, \sigma_{\text{color}})/f(0, \sigma_{\text{color}}), \tag{26.10}$$

After this preprocessing, an analysis of the attributes of the images is carried out, and based on these estimated results, feature maps for the current contents of the input window are constructed.

When forming the trajectories of medical images' observation, the following factors influencing the translation of the view:

(1) The color gradient values at points calculated by the Scharr operator [15] for a number of scales: 1, 1/4, 1/16, 1/64.

$$\begin{bmatrix} +3 & +10 & +3 \\ 0 & 0 & 0 \\ -3 & -10 & -3 \end{bmatrix} \begin{bmatrix} +3 & 0 & -3 \\ +10 & 0 & -10 \\ +3 & 0 & -3 \end{bmatrix}, \tag{26.11}$$

The Scharr operator was chosen because of its invariance to image rotation.

(2) The closeness of the point to the mobile center of attention, defined as the midpoint of the positions of the preceding points of window fixation. The map contains the normalized values of the Gaussian function with the mean located at the point of the center of attention.

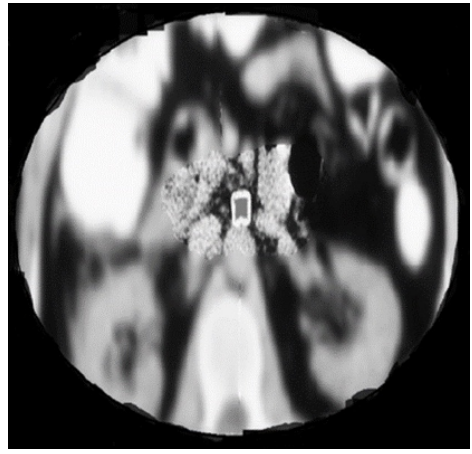


Figure 26.18. The image area processed by the model input window.

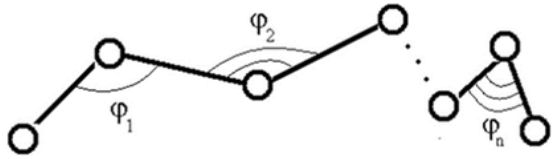


Figure 26.19. Angular characteristics of a trajectory. Circles denote fixation points, lines denote "saccades", φ_n denote the angle between two "saccades".

(3) Prevent the input window from returning to previously viewed areas of the image. The map contains nonzero values in places corresponding to the areas of the image where the model input window has already been fixed. At each next step of the input window these values decrease.

All operations with colors are performed in the color space CIE L*a*b*.

After normalizing the obtained feature maps, summing and eliminating the possible noise on the final map, the region with the highest values became the next fixation of the input window.

The obtained viewing trajectories were compared quantitatively using two algorithms and a combination of them to evaluate different properties of the viewing trajectory, namely:

(a) "How" to observe — angular characteristics of the trajectory.
(b) "Where" are the points of "fixation" of the trajectory located at the image.

As a spatial characteristic of the trajectory elements (Figure 26.19), the value of the angle φ_n, formed by two "saccades" (preceding this fixation and following it) was determined. Then, the Pearson correlation coefficient was calculated between the distributions of the φ_n for medical images of different classes, separated into 10 intervals with a step of 18°.

When examining an image of a person, so-called attention maps are formed, which contain information about which regions of the image attracted attention and which did not. Attention maps are built using information on the coordinates of visual fixations: fixation points are blurred, resulting in the formation of areas of attention. The more fixation points are located on the image, the area of attention is more significant and larger in size. Similarly, attention maps

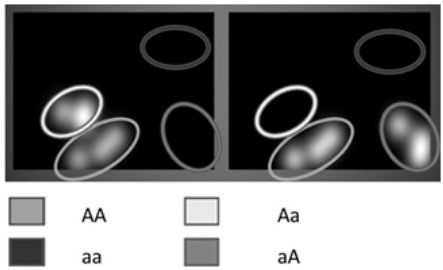


Figure 26.20. An example of correspondence between two attention maps.

were constructed for artificial trajectories of the observation using a model, after which these maps were compared.

To compare model attention maps, the Matthews correlation coefficient is used [26]:

$$M = \frac{AA \times aa - Aa \times aA}{\sqrt{(AA + Aa)(AA + aA)(aa + Aa)(aa + aA)}}, \qquad (26.12)$$

where AA is the number of selected points of the attention map constructed on the basis of one trajectory that match the selected points of the map constructed on the basis of the second trajectory;

aa is the number of non-selected points matching on both attention maps;

Aa and aA — the number of points selected on the first map of attention, but not selected at the second, and vice versa.

Figure 26.20 shows an example of maps of attention comparing.

For a more accurate image classification result, both trajectory characteristics were taken into account. For this purpose, an average value is calculated on the two correlation coefficients obtained as a result of the operation of the two algorithms presented above for the same pair of images and are within the range of -1 to 1.

To test the developed algorithm, a base was made that included X-ray images of the brain, chest, lung, foot, and pelvis from the compiled X-ray image database of a totally of 150 images (http://zhuravlev.info/, http://medimage.ru/).

For the images of the base, trajectories of their "observation" were obtained via the developed software model (Figure 26.21).

The correlation coefficients obtained using algorithm based on the angular characteristics within the classes were ranged from 0.65 to 0.75 (Diagonal elements in the Table 26.1). For different classes, the

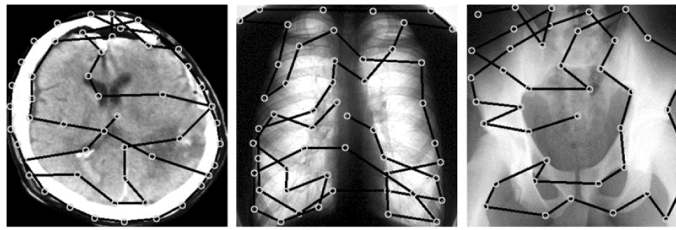


Figure 26.21. Viewing trajectories constructed using a modified behavioral vision model.

Table 26.1. Results of applying combination of two algorithms of viewing trajectories comparison.

Medical image type	**Lungs**	**Cerebrum**	**Rib cage**	**Pelvis**	**Foot**
Lungs	0.77 ± 0.03				
Cerebrum	0.1 ± 0.02	0.88 ± 0.05			
Rib cage	0.1 ± 0.02	0.2 ± 0.015	0.89 ± 0.01		
Pelvis	0.02 ± 0.001	0.02 ± 0.04	0.37 ± 0.02	0.69 ± 0.1	
Foot	0.3 ± 0.01	0.01 ± 0.03	0.21 ± 0.02	0.23 ± 0.05	0.62 ± 0.1

correlation coefficient was less than 0.5 (Non-diagonal elements in the Table 26.1). At the same time, the Mathews correlation coefficient, calculated via comparing of attention maps, resulted in values from 0.6 to 0.8 for images of the same class and less than 0.45 for different classes.

The table shows the results of the joint use of both algorithms. The table contains the average values of the correlation coefficients of the images of the same (diagonal table elements) and of different classes with standard deviations. It can be seen that the combination of the two algorithms is more effective for classification than using them separately. Namely, the correlation coefficient for images of the same class ranges from 0.62 to 0.89, while for images of different classes it does not exceed 0.3. The probability of a correct determination if two images belong to the same class was 92.8%, and detection of different classes — 86.7%.

As the investigation showed, the modified behavioral model of the visual perception (BMV) allows classifying medical images with high accuracy. It was found that the trajectories of viewing of similar images

constructed using the model have a similar appearance, while trajectories of viewing within different classes sharply differ from each other.

26.6 Conclusion

Active vision is an approach to describe visual perception as a purposeful behavior. It appeared in 1980 as a universal model of mechanisms and functions of visual perception in technical systems and living beings. Bidirectional knowledge transfer from technical to natural research and vice versa has led to significant progress in both areas. In particular, several hardware and software solutions which emulated natural vision were developed within active vision approach both for fundamental research and various applications. One such solution was the foveal sensor which simulates spatial and temporal characteristics of retina. It was implemented in software, in hardware, and even as a component of robotic platforms performing purposeful activity. An example of backward methodological enrichment (from a technical solution to fundamental research) is a foveal sensor-based model of viewing trajectories which was used to analyze and classify human gaze trajectories. Active vision concepts and implementation spurred the development of families of spatially heterogeneous (retina-like) descriptor extractors (DAISY, FREAK, BRISK, etc.) which were effectively used in different recognition tasks. In 2010s "manually" designed descriptors were replaced by embedded features generated by deep learning networks which demonstrated higher performance in a variety of scenarios. Nevertheless, retina-like sensors and active vision strategies of recognition are still of high interest wherever a large training set (required within deep learning approaches) is not available. Furthermore, active vision is both a family of methods for image analysis and a complex theory comprising many cognitive and perception layers. In particular, it specifies not only "how" to recognize objects in images, but also "what" to recognize or detect depending on observer's state and tasks. Active vision model aggregates high-level intellectual processes like decision-making and task prioritization, perceptive processes, and motor processes; thus, it provides wide opportunities for its use in real-life applications, where dynamic interaction with environment and humans may occur, first and foremost, in mobile robotics.

Acknowledgments

This research was supported by the Strategic Academic Leadership Program of the Southern Federal University ("Priority 2030").

References

[1] Alahi, A., Ortiz, R., and Vandergheynst, P. (2012) FREAK: Fast retina keypoint. *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, Rhode Island, Providence, USA, pp. 510–517.

[2] Barabanschikov, V.A. (1997) *Oculomotor Structures of Perception* (Publishing house "Institute of Psychology, Russian Academy of Sciences", Moscow) (in Russian).

[3] Belongie, S., Malik, J., and Puzicha, J. (2002) Shape matching and object recognition using shape contexts. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24(24), 509–521.

[4] Benoit, A. and Caplier, A. (2005) Biological approach for head motion detection and analysis. *Proceedings of 13th European Signal Processing Conference EUSIPCO-2005*, Antalya, Turkey.

[5] Favorskaya, M.N. (2009) Recognition of dynamic images based on predictive filters. *Bulletin of the Academician M.F. Reshetnev Siberian State Aerospace University*, 1 (in Russian).

[6] Ferman, A., Tekalp, A., and Mehrotra, R. (2002) Robust color histogram descriptors for video segment retrieval and identification. *IEEE Transactions on In Image Processing*, 11(5), 497–508.

[7] Fu, K.S. (1974) *Syntactic Methods in Pattern Recognition* (Academic Press, New York).

[8] Geusebroek, J.M., van den Boomgaard, R., Smeulders, A.W.M., and Geerts, H. (2001) Color invariance. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 23(12), 1338–1350.

[9] Gibson, J. (1988) *Ecological Approach to Visual Perception* (Progress, Moscow) (in Russian).

[10] Harris, C. and Stephens, M. (1988) A combined corner and edge detector. *Proceedings of the 4th Alvey Vision Conference*, pp. 147–151.

[11] Jähne, B., Scharr, H., and Körkel, S. (1999) Principles of filter design. *Handbook of Computer Vision and Applications* (Academic Press, Cambridge).

[12] Khrisanfova, E.N. and Perevozchikov, I.V. (2005) *Anthropology: Textbook*, 4nd Ed. (Science, Moscow) (in Russian).

[13] Kopeliovich, M.V., Kurdoglyan, A.V., and Petrushan, M.V. (2016) Method of dynamic patterns recognition by comparing trajectories

of descriptors in feature space. *Proceedings of Natural and Mathematical Science in the Modern World, SibAK, XLVII International Conference*, Novosibirsk, pp. 31–37 (in Russian).

[14] Koschan, A. and Abidi, M. (2005) Detection and classification of edges in color images. *Signal Processing Magazine, Special Issue on Color Image Processing*, 22(1), 64–73.

[15] Lampinen, Y., Shevchenko, M.A., Petrushan, M.V., and Samarin, A.I. (2005) Model of terrain map formation in the process of active interaction of the robot with the environment. *Proceedings of Neuroinformatics, 7th Scientific and Technical Conference*, pp. 155–164 (in Russian).

[16] Leutenegger, S., Chli, M., and Siegwart, R. (2011) BRISK: Binary robust invariant scalable keypoints. *Proceedings of IEEE International Conference on Computer Vision (ICCV)*, pp. 2548–2555.

[17] Lowe, D. (2004) Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision*, 60(2), 91–110.

[18] Matthews, B.W. (1975) Comparison of the predicted and observed secondary structure of T4 phage lysozyme. *Biochimica et Biophysica Acta*, 405, 442–451.

[19] Mikolajczyk, K. and Schmid, C. (2005) A performance evaluation of local descriptors. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 27(10), 1615–1630.

[20] Nayser, U. (1981) *Cognition and Reality* (Progress, Moscow) (in Russian).

[21] Noton, D. and Stark, L. (1971) Scanpaths in eye movements during pattern recognition. *Science*, 171, 308–311.

[22] Osinov, V.A., Shaposhnikov, D.G., Koltunova, T.I., and Podladchikova, L.N. (2012) Dynamics of spatial and temporal characteristics of image viewing: model and experiment. *Neuroinformatics*, 6(1), 1–11 (in Russian). http://www.niisi.ru/iont/ni/Journal/V6/N1/.

[23] Padma, A. and Sukanesh, R. (2011) Automatic diagnosis of abnormal tumor region from brain computed tomography images using wavelet based statistical texture features. *International Journal of Computer Science, Engineering and Information Technology (IJCSEIT)*, 1(3).

[24] Petrushan, M.V. (2010) Model of multi-level pre-tuning of the visual system for recognition of three-dimensional objects. *Journal Neurocomputers*, 2, 47–51 (in Russian).

[25] Petrushan, M.V. and Samarin, A.I. (2012) Method of contrasting descriptions of images of persons for the authorized access system. *Information Technologies*, 3, 74–78 (in Russian).

[26] Petrushan, M., Vermenko, Y., Shaposhnikov, D., and Anishchenko, S. (2013) Comparative analysis of color- and grayscale-based feature descriptions for image recognition. *Pattern Recognition and Image Analysis*, 23(3), 415–418.

[27] Podladchikova, L.N., *et al.* (2009) Model-based approach to study of mechanisms of complex image viewing. *Optical Memory and Neural Networks (Information Optics)*, 18(2), 114–121.
[28] Podladchikova, L., Shaposhnikov, D., Koltunova, T., Dyachenko, A., and Gusakova, V. (2009) Temporal dynamics of fixation duration, saccade amplitude, and viewing trajectory, *Journal of Integrative Neuroscience*, 8(4), 487–501.
[29] Rybak, I.A., Gusakova, V.I., Golovan, A.V., Podladchikova, L.N., and Shevtsova, N.A. (1998) A model of attention-guided visual perception and recognition. *Vision Research*, 35(15–16), 2387–2400.
[30] Rybak, I., *et al.* (2005) Attention-guided recognition based on "What"; and "Where" representations: A behavioral model. *Neurobiology of Attention* (Academic Press, New York), pp. 663–670.
[31] Samarin, A.I. and Shevchenko, M.A. (2002) Algorithm for automatically constructing descriptions of scene objects in the basis of the viewing movements of the foveal sensor. *Neuroinformatics — 2002* (MEPhI, Moscow), pp. 85–94 (in Russian).
[32] Sanderson, C. and Lovell, B.C. (2009) Multi-region probabilistic histograms for robust and scalable identity inference. Lecture Notes in Computer Science, Vol. 5558, pp. 199–208.
[33] Shaposhnikov, D.G., *et al.* (2001) The peculiarities of visual perception in the peripheral vision field. *Pattern Recognition and Image Analysis*, 11(2), 376–378.
[34] Tola, E., Lepetit, V., and Fua, P.D. (2010) An efficient dense descriptor applied to wide-baseline stereo. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 32(5), 815–830.
[35] Tomasi, C. and Kanade, T. (2004) Detection and tracking of point features. *Pattern Recognition*, 37, 165–168.
[36] Viola, P. and Jones, M. (2001) Rapid object detection using a boosted cascade of simple features. *Computer Vision and Pattern Recognition 2001*, pp. 511–518.
[37] Winder, S. and Brown, M. (2007) Learning local image descriptors. *Computer Vision and Pattern Recognition*, pp. 1–8.
[38] Wolfe, J.M., *et al.* (2005) Visual search for transparency and opacity: Attentional guidance by cue combination? *Journal of Vision*, 5, 257–274.
[39] Zelinsky, G.J. (2005) Specifying the components of attention in a visual search task. *Neurobiology of Attention* (Elsevier, Academic, Amsterdam), pp. 395–400.

© 2025 World Scientific Publishing Company
https://doi.org/10.1142/9789811267215_0027

Chapter 27

Implicit Modeling of Pronunciation Variation in Speech Recognition Systems

Vladimir Ya. Chuchupal

Federal Research Center "Computer Science and Control" of the Russian Academy of Sciences, Moscow, the Russian Federation
v.chuchupal@gmail.com

The variations in pronunciation of words in natural speech are one of the main sources of automatic speech recognition errors. This is the reason for the development and implementation of advanced pronunciation models in modern ASR systems. The examples of such variations include pronunciation variations that are caused by a fuzzy or an incomplete articulation that is frequently observed in spontaneous speech. The pronunciation variations techniques for automatic speech recognition that are known from the literature are considered. The evaluation results show that the observed effectiveness as a rule is far from the expected one. It is shown that of two existing frameworks to the building of pronunciation variation models, namely the explicit approach to pronunciation modeling and the implicit one, the latter is less developed but contains more possibilities for describing pronunciation variability. An implicit approach to pronunciation variation modeling for speech recognition systems is being developed that is implemented by means of smoothing of parameters of the adjacent acoustical sound models in phonemic transcription. It is proposed to use context-dependent smoothing so that the values of the smoothing parameters are conditioned by the current position and prosodic contexts of a sound. While

the pronunciation variation modeling approach that uses of combination of acoustical models has already been discussed in the literature, the method based on the context-depended smoothing of the adjacent models as far as we know has not been published yet. The experiments on the speech corporas that contained both the read and spontaneous speech showed the correctness of the proposal for the use of the context-dependent smoothing parameters which are conditioned by the features of phonemic context and prosody.

Keywords: Explicit pronunciation variation modeling, implicit pronunciation variation modeling, explicit approach to modeling the pronunciation variation, smoothing parameters of acoustic models, binary decision trees.

27.1 Introduction

Accuracy of automatic speech recognition can be significantly improved through the use of correct pronunciation models. This was confirmed in the course of simulation experiments [11, 16], in which the use of correct actual phoneme transcriptions of words resulted in a sharp decrease in the word error recognition rate, which decreased almost twice.

There are two main approaches to the modeling of pronunciation variation: explicit pronunciation variation modeling implies that pronunciation variation is reduced to word pronunciation variation, which, in turn, can be correctly represented by changing the basic phoneme transcriptions of words [20]. In this case, the model of pronunciation variation is associated with a set of pronunciation transcriptions of the word.

Implicit pronunciation variation modeling [16] describes pronunciation variation by changing the structure of acoustic models of sounds, leaving phoneme transcriptions unchanged. That is, there may be one phoneme transcription of a word, but of a complex nature, for example, in the form of a graph built from phonemes.

As a rule, industrial speech recognition systems use explicit pronunciation variation models, since they are simpler, easier to handle intuitively, more natural, and relatively simple to implement without significant costs in speech recognition systems. In addition, these models are simply interpreted as part of a statistical approach to speech recognition.

27.2 Modeling of Pronunciation Variation in Speech Recognition Systems

Let $X = \{x_t\}, t = 1, \ldots, T$ be a sequence for observations of speech parameters, and $W = \{w_i\}, i = 1, \ldots, N$ be a sequence of the corresponding words from a vocabulary. The most likely sequence of words W^* with given X values can be obtained from the following equation [7]:

$$\begin{aligned} W^* &= \underset{W}{\operatorname{argmax}}\, P(W|X) \\ &= \underset{W}{\operatorname{argmax}}\, \frac{P(X|W)P(W)}{P(X)} \\ &= \underset{W}{\operatorname{argmax}}\, P(X|W)P(W) \end{aligned} \tag{27.1}$$

Here, the first factor $P(X|W)$ in the numerator (27.1) denotes the likelihood of the data for a given sequence of speech parameters and is calculated based on the acoustic model. The probability of a word sequence $P(W)$ is determined using a language model. The denominator $P(X)$ is essentially a normalizing term.

Denote the phoneme transcription of some word w by t^w; the set of all phonemic transcriptions of this word as T^w. The set of valid phoneme transcriptions for a word sequence W is denoted by T^W. The notation t^W will be used for an arbitrary phoneme transcription from the T^W set.

Suppose that for each word w a set of its valid transcriptions T^w is already known, along with their probabilities.

Consider equality:

$$\begin{aligned} P(X|W) &= \sum_{t^W \in T^W} P(X, t^W|W) \\ &= \sum_{t^W \in T^W} P(X|t^W, W)P(t^W|W). \end{aligned} \tag{27.2}$$

Probabilities $P(t^W|W)$ are estimated using the explicit model of pronunciation variation. The model parameters are phoneme transcriptions from T^W along with their relative frequencies $\{P(t^W|W), t^W \in T^W\}$.

Acoustic parameters are actually determined by phoneme transcription and do not depend on the word sequence, i.e.

$$P(X|t^W, W) = P(X|t^W).$$

Then it follows from equation (27.2) that

$$P(X|W) = \sum_{t^W \in T^W} P(X|t^W)P(t^W|W)$$

and equation (27.1) can be written as

$$W^* = \underset{W}{\operatorname{argmax}}\, P(W) \sum_{t^W \in T^W} P(X|t^W)P(t^W|W) \tag{27.3}$$

which in practice, approximating the sum of the terms by its max term (i.e. using the Viterbi approximation) can be interpreted as:

$$W^* = \underset{W}{\operatorname{argmax}}\, P(W) \max_{t^W \in T^W} P(X|t^W)P(t^W|W). \tag{27.4}$$

Conditions (27.3) and (27.4) differ from equation (27.1) by the presence of term $P(t^W|W)$, which takes into account the pronunciation variation of the words from sequence W. The set of probabilities $P(T^W|W) = \{P(t^W|W), t^W \in T^W\}$ are parameters of this model (in the same way as the set of valid transcriptions t^w of each word.

To recognize speech using equation (27.4), it is necessary to know the parameter values for three models: acoustic, pronunciation, and language models.

The estimation of the parameter values by the method of maximum a posteriori probability corresponds to the use of the following criterion:

$$P(W|X) = \underset{t^W}{\operatorname{argmax}} \frac{P(X|t^W)P(t^W|W)P(W)}{\sum_{t^W \in T^W} P(X|t^W)P(t^W|W)P(W)}. \tag{27.5}$$

The resulting values of the parameters can be viewed as a discriminant solution (27.5) in the sense that it maximizes the probability of correct (for training data) models while minimizing the total probability of all possible models.

The parameters of the language model $P(W)$ in equation (27.5), as well as for equation (27.3), can be considered independent of the training acoustic signals, and their evaluation should be performed separately on the textual data corpus. However, the parameters of

the pronunciation models $P(T^W|W)$ depend on the acoustic training data, therefore their estimate independent of the parameters of the acoustic models is incorrect.

Practical use of the equation (27.5) was rather difficult until recently since it implies the availability of large data, and so far only a few similar experiments are known [8].

In this situation, it is much easier to estimate the parameters of the pronunciation model by the maximum likelihood method, i.e. using only the numerator of equation (27.5).

Suppose that the training corpus of speech data X is such that for all utterances not only the word sequence $W = w_1 w_2, \ldots, w_N$ is known, but also their phonemic transcriptions $t^W = t_1 t_2, \ldots, t_N, t_i \in T^{w_i}$.

In this case, the maximum likelihood estimate of parameters $p(t_i|w_i\}, i = 1, ..N$ is determined from the following conditions:

$$\max \prod_{i=1}^{N} p(t_i|w_i), \tag{27.6}$$

$$\forall v : \sum_{t_i : w_i = v} p(t_i|w_i) = 1. \tag{27.7}$$

That is, as an experimental frequency of the appearance of the corresponding model in the data corpus:

$$p(t^w|w\} = \frac{\#\{t^w\}}{\#\{w\}}, \tag{27.8}$$

where # denotes the number of events in curly brackets encountered in the training data.

Note that the frequency parameter estimates equation (27.6) were obtained under the assumption that the composition of the phoneme transcriptions of words was determined. Since phonemic transcriptions can be selected in various ways, and the parameters of the pronunciation and acoustic models depend on each other, their independent estimation will be incorrect.

In practice, parameters of the model are determined separately, one by one. For example, word transcriptions and values of their frequencies are set up initially using knowledge. Then based on the results of the data corpus recognition, the frequencies of word transcriptions are estimated using equation (27.8). Rare variants of word transcriptions are discarded and acoustical models are

re-estimated using the updated word transcriptions. Then data corpus recognition step is repeated and the frequencies of word transcriptions are re-evaluated.

27.3 Modification of Speech Recognition Procedures to Account for Pronunciation Variation

The straightforward way to implement the pronunciation variation model in speech recognition is based on a simple replenishment of the vocabulary with new pronunciation variants. In this case, instead of equation (27.3), the following condition is used:

$$W^* = \underset{W}{\operatorname{argmax}} P(W) \sum_{t^W \in T^W} P(X|t^W) \tag{27.9}$$

which determines the most likely sequence of phonetic transcriptions of words.

When vocabulary words have one single transcription, solutions (27.3) and (27.9) are obviously equivalent, but with more than one pronunciation, condition (27.9) becomes non-optimal for finding the most likely sequence of words.

From the previous section it follows that the most likely sequence of words W^* can be determined from the equation (27.3)

The recognition algorithm using criterion (27.3) differs from the version for (27.9) in that it is necessary to take into account the probability of individual transcriptions of the word.

The condition (27.4) implies the suboptimal variant of the estimate, replacing the sum of the likelihoods of the models with the maximum likelihood.

27.4 Effectiveness of Using Explicit Pronunciation Variation Models

The implementation of explicit pronunciation models is reduced to solving three problems:

- identification of likely pronunciation variants for vocabulary words,
- evaluation of the empirical transcription frequencies,
- use of pronunciation variants when recognizing words.

Variants of word pronunciation, in turn, can be obtained through the following:

- phoneme recognizer applied to the data corpus [6],
- by modifying basic transcriptions using expert knowledge [8, 20],
- by modifying basic transcriptions using rules derived from machine learning techniques.

The most obvious way to select phonemic transcription variants automatically is based on the use of the phoneme recognizer. However, in practice, this is often inefficient due to the insufficient accuracy of the recognizer.

In [6], in large data, actual phoneme transcriptions were determined with phoneme-by-phoneme recognition, and their probabilities were estimated using an interpolated model that included empirical and parametric frequencies. The parametric frequency was determined by the probabilities of the transition of canonical phonemes to surface ones. As a result, for a linear combination of empirical and parametric models, word error rate decreased from 34.8% to 33.0%, or by 5.2%, in relative value.

The use of variants of pronunciation transcriptions obtained based on the application of rules both based on knowledge and derived from data in [20] on the VIOS corpus material (for the Dutch language) reduced the word error rate to ∼0.8% (from the initial 10.7% to 9.9%), with a significant (4.9 per word on average) number of valid pronunciation variants.

In [2], decision trees were constructed on the data corpus for probabilistic mapping of canonical phonemes into surface ones. These trees were then used either to generate pronunciation variants of the vocabulary words or to generate pronunciation variants of full phrases. This resulted in lowering the word error rate by 2% (with 41% of the initial value).

Using discriminant methods (minimizing the word error rate) to calculate the weights of alternative pronunciations and search with summation of the weights of alternative pronunciations in [18] provided a gain in word error rate of 2.2%. As a pronunciation variation feature, the rate of speech was also considered in this work.

Lehr *et al.* [8] proposed and investigated a discriminant model of pronunciation variation for a dialect, namely African English.

Initially, expert methods were used to find contextual rules for generation of possible pronunciation variants using basic transcriptions, and then using the so-called transcriptions from the oracle (i.e. the most correct ones from the list of N best in recognition) the neural network learned to assign weights to the rules, which then made it possible to enter weights for the pronunciation variants of words derived from these rules.

Since the presence of a dialect, as a rule, correlates with syntactic deviations, language models were also adapted in this work, along with phoneme transcriptions.

The theoretically possible improvement in the word error rate (for oracle transcriptions) in this experiment was more than 10%: from 38.2% to 28.1%. In fact, the introduction of pronunciation models reduced an error rate by 2.1%, while the main contribution to this improvement was made by the adaptation of language models, and the pronunciation variation models yielded 0.6%. A similar model for optimizing the transcriptions of the American standard language also made it possible to reduce the word error rate by 0.7% (from 25.2% to 24.5%).

Despite the fact that the references describe many other ways of overcoming the presented problems through the use of suboptimal, discriminant or non-parametric approaches, the gain in recognition accuracy that is observed when using explicit models is not at all what one would expect based on the results of simulation experiments [11].

27.5 Effectiveness of Using Explicit Pronunciation Models for the Russian Language

The proposed pronunciation variation models were compared in a numerical experiment on the ISABASE 2 [1] and TeCoRus [3] data corpora.

The training sample consisted of utterances from 200 ISABASE-2 speakers (20 hours of speech) and 50 TeCoRus speakers (2.5 hours). The average number of variants of pronunciation transcriptions per word was estimated as $\sum_{i=1}^{N} t_i/N$ where N is the number of words in the vocabulary, and t_i is the number of pronunciation transcriptions for the ith word. As a vocabulary for which the average variation was calculated, numbers and numerals were considered. For the first

Table 27.1. Word error rate (WER) for pronunciation variation models, numbers and numerals, Russian.

Method	Basic	Simple	Opt	SubOpt
TeCoRus read speech	1.6	5.8	2.00	3.3
TeCoRus free speech	7.8	7.6	7.4	7.4
PronExRu conversational speech	69.3	57.4	59.7	62.1

test, the variation of the digital vocabulary was equal to 1.9, for the second and third tests, the vocabulary of numbers and numerals had a variation value of 1.3.

The material of the first test consisted of TeCoRus data: 776 continuous numeric sequences, from 2 to 16 digits in length, a total of 3147 digits, the vocabulary consisted of word forms of digits. The pronunciation variants were evaluated on the ISABASE-2 material, the average number of pronunciation variants was 1.9. To evaluate the effect of only acoustic characteristics on recognition accuracy, the language model was disabled.

The second experiment was conducted on a subset of the TeCoRus corpus, which included all the material of the first test, and free speech with numerical information additionally. These were answers to questions that contained birth dates, zip codes, school numbers, etc., 867 additional statements from 11 speakers in total. The vocabulary included 129 word forms, including 26 numbers and numerals. The average number of transcriptions was 1.3.

The third experiment was conducted for natural spoken language. Data for testing, a subset of the PronExRu [13] corpus consisted of fragments of audio interviews from 10 speakers taken from the Internet. These audio fragments were converted from MP3 format to WAV format. The interviews were previously automatically segmented, and fragments were selected that contained numbers and numerals. The test sample consisted of 200 short utterances, with a vocabulary, including a subvocabulary of numerals with pronunciation variants used in the second test.

The results of the numerical experiment expressed in terms of the word error rate are given in Table 27.1.

The column “Basic” contains the results for the case of using only basic phoneme transcriptions. The column “Simple” corresponds to

the method of accounting for variation using equation (27.9), the column "Opt" — to the method of accounting for variation based on equation (27.3), and the column "SubOpt" — to the method based on equation (27.4).

The results given in Table 27.1 for read speech can be interpreted as evidence of the absence of variation in the pronunciation of numbers in this corpus. This can be explained by the fact that the TeCoRus speakers lived in Moscow, had higher educations, and belonged mainly to two professional groups (including linguists) who carefully read the proposed digital material.

The results given in Table 27.1 for free speech were more expected: the use of the method of frequency weighting for pronunciation variants equation (27.3) turned out to be optimal for minimizing the word error rate. The method of simple addition of transcriptions equation (27.9) turned out to be less effective compared to both optimal and suboptimal equation (27.4) ones that take into account the frequency of transcriptions but is still preferable to using only canonical models.

At the same time, changes in word error rate as a result of using pronunciation models seem insignificant.

The significantly higher level of absolute recognition error in the line for the Interview is caused by the spontaneous nature of conversational speech, the use of the MP3 codec, the presence of external noise, and the disabling of the language model. Unlike previous tests, where the decrease in the relative level of error due to the use of explicit models did not exceed 5%, in this case, for all methods of variation modeling, there is a substantial decrease in the level of errors to more than by 11%.

Although there is no substantial difference in the results for model equations (27.3), (27.9), and (27.4), the optimal method equation (27.3) was not the best in this test. This effect can be explained. In the test material used, numbers have a much higher frequency of occurrence than other words. In the absence of the language model, method equation (27.9) allows taking into account the high frequency of numbers, partially replacing the unigram language model. Weighing phoneme transcriptions equation (27.3) equalizes the frequency of occurrence of words, which does not correspond to the actual statistics of data.

27.6 Implicit Pronunciation Models

The main idea of the explicit approach to modeling the pronunciation variation is that all pronunciation changes can be quite accurately represented by changes in phoneme transcriptions of words, that is, by replacing, inserting, or deleting phonemes of basic transcriptions. Experimental analysis of speech data shows that pronunciation variation, especially in the case of conversational speech, can be represented more fully using models that can describe partial changes in the acoustic properties of the basic transcription phonemes. Such an approach to modeling the pronunciation variation is called implicit [17].

Unlike the explicit approach, implicit pronunciation variation models are implemented by constructing more complex acoustic models of phonemes, in basic phoneme transcription of words, for example, by replacing hidden Markov models from left to right with models in the form of networks of states.

Implicit pronunciation models are implemented by modifying acoustic models of basic transcription sounds, for example:

- as a network of Markov models [5],
- as linear combinations of states of Markov models [9] or by mixing probability densities [16, 17],
- using matrices of articulatory features [10],
- using larger acoustical units: syllables, words [19].

Hain and Woodland [4] proposed a dynamic model selection technique to take into account pronunciation variation in spoken language. Machine-learning methods made it possible to build a probabilistic mapping of basic phonemes into acoustic models of observed sounds. As a set of acoustic models of the observed sounds, models of triphones from basic transcriptions were used. Thus, each phoneme in the text was represented as a model of several parallel triphones, each of which had its own weight. In the Switchboard spoken language corpus, the word error rate decreased by 0.77% in absolute or 1.7% in relative (45.04% and 44.27%) units.

In [9], the pronunciation variation is proposed to be modeled using interpolated models. For this, it is proposed to use the so-called partial change phone models (PCPMs). Each such model corresponds to a pair of phonemes of type $a : b$, for the situation when in the

position of base phoneme a phoneme b is actually pronounced. Since in practice there are lots of variants of such replacements, for conditions with a limited sample of training data, a clustering algorithm is proposed for matching the states of Markov models $a : b$ and the states of triphone models with a central phoneme a. The use of such a model for recognizing spoken Chinese language resulted in a gain in a syllable recognition error of 2.39% (from the initial error of 42.23% to 39.84%).

In [16, 17] pronunciation variation is modeled by using acoustic models of sounds obtained from the states of the hidden Markov models of canonical and surface (actually pronounced) phonemes. These models can either be used simultaneously [17], or by combining densities [16] with re-evaluation of their weights. In experiments on the Switchboard data corpus [17], due to the use of such combined models, the word error rate decreased by 1.7% (from 39.4% to 37.7%).

Pronunciation variation is relatively easily interpreted in terms of the theory of speech generation as a result of desynchronization of articulatory movements. Livescu and Glass [10] showed that most of the effects associated with a change in pronunciation (nasalization, partial replacement, or change in order) of sounds that are difficult to represent as changes in the phoneme transcription of words can be explained by asynchronous changes in articulatory features.

In this case, the phoneme transcription of words is replaced by a matrix of the values of articulatory features, and a Bayesian dynamic model is used to describe parallel asynchronous flows of features.

The possibility to model pronunciation of words by selection using data corpus, the number of acoustic states, and alternative phonemic transcriptions was investigated in [19]. The selection of the specified transcription parameters for the word is determined on the basis of machine learning; the resulting acoustic transcriptions are word-dependent. The use of such a model in the data corpus of "Spoken LAnguage Car Control" task resulted in the reduction of the word error rate by about 2.5%.

In most of the listed works, both for explicit and implicit approaches to variation modeling, the parameters of surface sound models are evaluated by statistical methods regardless of the contextual information other than the phonetic context of the sound. It is

known, however, that a positive effect can be achieved through models that take into account the influence of the semantic and syntactic context.

In [12], the pronunciation changes due to the supposedly syntactic and grammatical context were studied using the Switchboard corpus material. The authors built a phoneme-to-phoneme type converter that determined the most likely surface phonemes from a canonical phoneme transcription. In addition, syntactic and grammatical features such as part of speech, position within a sentence, stress, type of dialogue gesture were used as context features. Indirect experiments showed that the discrepancy between the predicted and expert transcriptions, in terms of the weighted phoneme error, as compared to the case where only the left and right phoneme contexts are used in the phoneme-to-phoneme conversion, decreased by 5.8%: from 25.2% to 19.4%.

Accounting for speech rate is obviously also useful for modeling pronunciation variation. In the NIST 2000 Hub-5 data corpus, the use of variation models to account for speech rate resulted in a reduction in the word error rate by 2.2%: from 54.6% to 52.4% [21].

Thus, the gain in recognition quality from the use of implicit pronunciation variation models does not differ significantly from the gain obtained when using explicit models and in both cases is significantly less than one would expect [5].

A possible reason for such very moderate improvements is that almost all the considered approaches, with the exception of speech rate estimation methods, use very limited contextual information. Results are known [12] that demonstrate that pronunciation variation is of a substantially conditional nature. These results can be interpreted as the ability to predict the level of pronunciation variation of words based on the assessment of the actual prosodic, syntactic, and semantic contexts.

27.7 Modeling Pronunciation Variation by Smoothing Parameters of Acoustic Models

Let m and n denote sound models, $P(x|m), P(x|n)$ denote conditional probabilities for observing speech parameters x for given models. Then model m which is smoothed by model n, $P(x|m, n)$ can be

defined similarly [9]:

$$P_\lambda(x|m,n) = \lambda P(x|m) + (1-\lambda)P(x|n). \tag{27.10}$$

Expression (27.10) makes it possible to describe, albeit in a simplified form, some pronounced changes that are often observed in conversational speech. For example, a value of λ equal to 0 means that sound m is completely replaced by sound n. Similarly, $\lambda = 0.5$ describes the situation of an incomplete replacement of the phonetic quality, which (in a simplified way) corresponds to the effects of nasalization, voicing, or devoicing sounds in conversational speech.

Let sound m be observed over the time interval $s(m), \dots e(m)$ with the corresponding parameters $x_{s(m)}, \dots x_{e(m)}$. The value of the smoothing coefficient λ in (27.10) can be found similarly to estimating the weights of mixtures when a model learns to account for Gaussian mixture [15]:

$$\lambda_{m.n} = \frac{\sum_{t=s(m)}^{e(m)} P(x_t|m)}{\sum_{t=s(m)}^{e(m)} \left(P(x_t|m) + P(x_t|n)\right)}. \tag{27.11}$$

For a data corpus U, which consists of R utterances $U = \{u_r, r = 1, \dots R\}$, the value of $\lambda_{m,n}$ can be calculated by averaging local estimates (27.11) for all observations of phoneme pairs (m,n), similar to estimates of Markov model parameters [15]:

$$\hat{\lambda}_{m.n} = \frac{\sum_{r=1}^{R} \sum_{(m,n)\in u_r} \lambda_{m,n} P(m)}{\sum_{r=1}^{R} \sum_{(m,n)\in u_r} P(m)}, \tag{27.12}$$

where $P(m)$ is the likelihood of sound m when observing (m,n).

Let V be a vector of contextual features, that is, features of the phonemic, positional, prosodic, linguistic and syntactic context, the presence of which [12] correlates with the observed level of pronunciation variation:

$$\mathrm{V(c, l, r, nPh, pPos, ROS, wPoS, POS, LM)}:, \tag{27.13}$$

where "c": indicates the target phoneme, "l" and "r": its left and right phoneme context, "nPh": subsequent phoneme, "pPoS": position of the phoneme in the word, "ROS": rate of speech, "wPOS": word

position in the phrase, "POS": part of speech, "LM": meaning of the language model for the word.

Let us experimentally verify the assumption that the level of pronunciation variation described by the value of $P(\lambda|V)$ really depends on the features (27.13).

27.8 Numerical Experiments

The proof that sound models (27.10) using interpolated parameters can significantly reduce the level of errors is best obtained on the basis of a speech recognition experiment. However, such an experiment requires the implementation of interpolated models in the training and recognition procedures. Preliminary experiments that prove the effectiveness of the proposed models, albeit indirectly, can be implemented by estimating the parameter λ on the data corpus to show that the parameter value depends on context features significantly (27.13).

The speech material of TeCoRus, RuSpeech, and PronExRu was divided into three samples designed for training, tuning, and testing. The training sample consisted of speech material from the RuSpeech and TeCoRus corpora, which contained read speeches from 200 speakers. The tuning data were used to estimate the values of the interpolation parameters. They consisted of the test material of the RuSpeech corpus, a total of 1,000 utterances from 10 speakers. Finally, the test material consisted of PronExRu data (200 utterances, 9 speakers).

The training data was used to create gender-oriented acoustic models. The evaluation of the feature values in (27.13), with the exception of the ROS (speech rate) indicator, was performed in two stages using the results of automatic speech recognition obtained in the first stage.

The ROS parameter was estimated using the algorithm described in [21]. To reduce the computational complexity, a sampling procedure was used instead of an exhaustive search for possible combinations of sound durations.

In the course of preliminary experiments, discrete values of features (27.13) were estimated for each central state of Markov models

of allophones. Then the values of λ parameters were calculated using equation (27.12).

In order to be able to rank features, two binary decision trees were constructed. One tree was constructed using the read TeCoRus speech material. The other tree was constructed using the conversational speech of the PronEx data corpus. The set of questions for "growing" the trees contained all sorts of questions about the presence and values of features (27.13) for the current Markov model state (for example, "Is this state a part of noun?", "Is this state a part of the beginning of a word?"); apart from this, the questionnaire included a set of standard questions that are used in the construction of phonemic binary decision trees for the synthesis of the alphabet for context-dependent states of Markov models of allophones. 82 questions in total were used: 30 questions related to features from (27.13) and 52 questions related to the phoneme context of the sound and its phoneme quality.

As a criterion for splitting nodes when "growing" the trees, the magnitude of the information was used [14], and the change in entropy when the parent node is divided into two child nodes after using the question:

$$\Delta H_\lambda(t, f) = H_\lambda(t) - \frac{|t_+|}{|t|} H_\lambda(t-) - \frac{|t_-|}{|t|} H_\lambda(t+) \qquad (27.14)$$

Here $H_\lambda(t), H_\lambda(t-)$ и $H_\lambda(t+)$ denote the entropy value for the data in the parent node t and the entropy of the data in the child nodes t_- and t_+ formed after the binary separation of the parent node t.

In order to numerically assess the importance of each feature, its weights were estimated. For each given feature over all the nodes of the tree, where the question about this feature was selected as the best separator of the vertex, the values of the magnitude of the information gain were summed up. That is, the total change in gain equation (27.14) was calculated for this feature. Then these total changes were normalized by the value of the maximum value to 100 (i.e. the most important feature has a weight of 100):

$$I(f) = \sum_{t \in T} \Delta H_\lambda(t, f) \qquad (27.15)$$

where summation is performed over all nodes of the tree.

The result of constructing these trees is contained in the following Table 27.2, which contains a list of the most important

Table 27.2. Most important features and their weights.

Feature	Importance in validation data	Feature	Importance in test data
POS-noun	100	RATE-F	100
RATE-m	72	L-Labial	70
L-soft	55	STRESS	63
RATE-F	42	POS-noun	54
R-VoiceLess	40	L-Vow	52
L-Sonant	37	PpMdl	51
R-Forv	35	R-Labial	42
L-Forv	31	NEXT-VOW	39
POS-adj	30	RATE-m	39
R-Sil	29	R-Forv	35
STRESS	29	R-Forw	29

features for the tuning and test parts of the data corpus. Abbreviations in Table 27.2 have the following meaning: "POS-noun" means "noun", "POS-adj" means "adjective", "RATE-F" means "fast rate of speech", "RATE-m" means "moderate rate of speech", "STRESS" means "stressed sound", "PpMdl" means "sound inside of the word", "R-sil" means "sound after a pause", "L-soft" means "the previous sound is a soft consonant", "R-VoiceLess" means "sound before a pause", "L-Sonant" means "the previous sound is a voiced consonant", "R-Forv" means "next sound is a front consonant", "R-Forw" means "next sound is a front vowel", "NEXT-VOW" means "next sound is a vowel", "R-Labial" means "next sound is a labial consonant", "L-Forv" means "the previous phone is a front consonant".

The values of the smoothing parameter λ measured by (27.11) for most of the time were in the range from 0.2 to 0.8, and the likelihood value of the data for the smoothed models was higher than with the initial acoustic models.

As follows from Table 27.2, the most important feature for read speech material (RuSpeech corpus) is a feature of a part of speech. For conversational speech (PronEx corpus), the most important feature is the feature of rate of speech. Both of these features relate to the proposed set of features for pronunciation variation (27.13). Of the eleven most important features for the read speech material, six features are included in the set of features. For the conversational

speech material, there are five such features. If one considers the most important features weighing more than 50, then there are only three such features for read speech and two of them belong to the set (27.13). For conversational speech, there are six such features, four of which are from the set (27.13).

The most important features for both data corpora are those associated with the rate of speech, part of speech, and the position of the sound relative to the stress.

Based on these preliminary results, it can be argued that the value of λ determined in accordance with (27.12) really depends on the values of the features listed in (27.13). The proposed features of pronunciation variation based on prosodic and positional contexts of the word and sound correlate with the observed changes in acoustic quality, i.e. with pronunciation variation. Thus, model equation (27.10) based on smoothing the parameters of neighboring acoustic models, can be used to take into account the effects of pronunciation variation in the recognition of natural speech. Such a model can be implemented, for example, by post-processing, as a second phase in recognizing.

27.9 Conclusion

This chapter proposes an implicit pronunciation variation model as a way to increase the accuracy of automatic speech recognition. Pronunciation changes in spoken language are proposed to be taken into account through the use of acoustic models of sounds with smoothed parameters so that the value of the parameters that regulate the level depends on the contextual and positional features of the sound and the word in which the sound is located.

A set of several potential context-dependent features of variation is proposed and their value as variation predictors is studied numerically. To do this, decision trees were constructed on the basis of the material of read and conversational speech, using a questionnaire that contained both standard questions for constructing phonetic decision trees and questions relating to potential features of variation. All features were ranked by importance according to the metric of information gain.

It was found that a number of proposed context-dependent features of variation actually correlate with the observed pronunciation

variation, exceeding in importance most of the common features of the phoneme context of the sound. Thus, the proposed model based on parameter smoothing can be used to account for the effects of pronunciation variation in natural speech recognition.

References

[1] Bogdanov, D.S., Krivnova, O.F., Podrabinovitch, A.J., and Arlazarov, V.L. (2004) Creation of Russian speech databases: Design, processing, development tools. *Proceedings of International Conference on Speech and Computers, SPECOM.*

[2] Byrne, B., Finke, M., Khudanpur, S., McDonough, J., Nock, H., Riley, M., Saraclar, M., Wooters, C., and Zavaliagkos G. (1997) Pronunciation modelling for conversational speech recognition: A status report from WS97. *Proceedings of IEEE Workshop on Automatic Speech Recognition and Understanding*, *ASRU*, pp. 26–33.

[3] Chuchupal, V.J, Makovkin, K.A., Chichagov, A.V., Kouznetsov, V.B., and Ogaryshev, V.F. (2005) Speech data corpus TeCoRus. Database registration certificate No ∼2005620205, Rospatent (in Russian).

[4] Hain, T. and Woodland, P.C. (1999) Dynamic HMM selection for continuous speech recognition. *Proceedings of International Conference on EuroSpeech,* pp. 1327–1330.

[5] Hain, T. (2005) Implicit modelling of pronunciation variation in automatic speech recognition. *Speech Communication*, 46, 171–188.

[6] Hitchinson, B. and Droppo, J. (2011) Learning non-parametric models of pronunciation in automatic speech recognition. *Proceedings of International Conference on Acoustics, Speech, and Signal Processing, ICASSP*, pp. 4904–4907.

[7] Jelinek, F. (1997) *Statistical Methods for Speech Recognition* (The MIT Press, USA).

[8] Lehr, M., Gorman, K., and Shafran, I. (2014) Discriminative pronunciation modeling for dialectal speech recognition. *Proceedings of Conference of International Speech Communication Association, Interspeech*, pp. 1458–1462.

[9] Liu, Y. (2003) Modeling partial pronunciation variations for spontaneous Mandarin speech recognition. *Computer Speech & Language*, 17(4), 357–379.

[10] Livescu, L. and Glass, J. (2004) Feature-based pronunciation modeling for speech recognition. *Proceedings of Human Language Technologies Conference of the North American Chapter of the Association for Computational Linguistics*, *HLT-NAACL 2004*, USA.

[11] McAllaster, D., Gillick, L., Scattone, F., and Newman, M. (1998) Fabricating conversational speech data with acoustic models: a program to examine model-data mismatch. *Proceedings of International Conference on Speech and Language Processing*, pp. 1847–1850.

[12] Ostendorf, M., Shafran, I., and Bates, R. (2003) Prosody models for conversational speech recognition. *Proceedings of 2nd Plenary Meeting and Symposium on Prosody and Speech Processing*, pp. 147–154.

[13] PronExRu. (2016) Database registration certificate No 2016620687, Rospatent (in Russian).

[14] Quinlan, J.R. (1986) Induction of decision trees. *Machine Learning*, 1, 81–106.

[15] Rabiner, L. and Biing-Hwang, J. (1993) *Fundamentals of Speech Recognition.* Prentice Hall Signal Processing Series (PTR, New Jersey).

[16] Saraclar, M., Nock, H., and Khudanpur, S. (2000) Pronunciation modeling by sharing Gaussian densities across phonetic models. *Computer Speech and Language*, 14(4), 137–160.

[17] Saraclar, M. and Khudanpur, S. (2004) Pronunciation change in conversational speech and its implications for automatic speech recognition. *Computer Speech and Language*, 18(4), 375–395.

[18] Schramm, H. (2006) Modeling spontaneous speech variability for large vocabulary continuous speech recognition, PhD Thesis, Technical University of Aachen, Germany.

[19] Spiess, T., Wrede, B., Fink, G.A., and Kummert, F. (2003) Data-driven Pronunciation Modeling for ASR using Acoustic Subword Units. *Proceedings of International Conference InterSpeech*, pp. 2549–2552.

[20] Wester, M. (2003) Pronunciation modeling for ASR — knowledge-based and data-derived methods. *Computer Speech and Language*, 17(1), 69–85.

[21] Zheng, J., Franco, H., and Stolcke, A. (2003) Modeling word-level rate of speech variation in large vocabulary conversational speech recognition. *Speech Communication*, 41(2–3), 273–285.

© 2025 World Scientific Publishing Company
https://doi.org/10.1142/9789811267215_bmatter

Index

A

active vision, 937
aerial imagery digitizing, 801
algebra
  Birkhoff/Lipson, 33–34
  Gurevich's descriptive image, 30–34
  graded, 2, 34
  Maltsev algebraic system, 34
  Ritter's standard image, 6, 30, 87
  superalgebra, 34
  Zhuravlev's algorithm, 28
algebraization of image analysis techniques, 7, 29
algorithm
  based on estimate calculation, 26
  based on estimate calculating excepting spatial data efficient recognition, 8
  boosted gready committee, 484
  deep learning, 940
  expectation-maximization, 323–360
  Levenberg-Marquardt, 754, 756, 758
  vessel tracing, 717
  Zhuravlev's algebra of algorithms, 28
analysis
  blood vessels digital, 699–725
  comparative image classes, 102
  descriptive image, 1–10, 14–15
  discriminant, 520, 522, 580, 594, 605–609, 707, 722–723
  linear discriminant, 520, 522, 594, 603
  principal component, 307, 314, 369, 520, 962
  spectral, 208, 218, 550, 557, 599, 618–619, 629
approach
  algebraic approach to the problem of synthesizing pattern recognition algorithms, 915
  bionic, 940
  descriptive approach to image analysis, 3–4, 10, 18, 24, 29–30, 36
  experiment-based, 510
  explicit approach to modeling the pronunciation variation, 985
  model-based, 511
  Pavel category, 27, 30
  quasi-model, 511
  structural, 639
  Zhuravlev's algebraic approach to recognition and classification problems, 27, 915–934
artificial neural networks, 505–535, 803, 866

artificial ultrasound slices, 735
audible stimulus, 560–563
autonomous determination of
artificial neural networks
parameters, 515

B

basis
Karunen-Loeve, 548
orthogonal, 547
restoring, 776, 778
smoothing, 776
binary decision trees, 990
blood vessel, 710
diameter measurement, 714
digital registration, 706

C

chaotic properties, 591
classification, 311
clinical investigation, 706
committee solution
generalized, 461
existence criterion, 466
computational complexity, 722,
724–725
computational geometry, 777
computer cartography, 556
contour
continuous elementary, 209, 216,
222, 224, 227–228
correlation-extreme method, 783
description, 557
random, 205
spectrum of a stationary random,
222
variance of a random, 206
cordial vascular system, 715
correcting operations, 921
count intensity, 327
cyclic variations, 695

D

data mapping error, 367, 376
deconvolution, 756
deep learning, 940
description
contour, 557
completeness, 955
obtaining consistent, 792
weakly formalized object, 795–796
descriptive algorithmic scheme of
image representation, 5, 7, 18
descriptive image algebra, 1–10,
31–40
interpretability of operations, 33
physical sense of the operation, 36
physically interpretable in the
strong sense operation, 37
physically interpretable in the
context of image analysis and
recognition operation, 37
physically interpretable in the
weak sense operation, 38
visually interpretable operation, 37
descriptive logic transformation class,
24
descriptor, 181, 834, 853, 938–939
detection
blood vessel, 709
epileptiform activity, 658
object, 123
ophthalmologic disease, 701
seizure, 658
spindles, 658
diabetic retinopathy, 700
distance
Hamming, 550, 853, 919
Harsdorf, 795
dot matrix, 551, 566, 568
dynamic programming, 569, 836,
861–865, 881

E

electroencephalogram, 616
electroencephalography, 657–661, 663,
667–670, 672–674
element, 104
elevation sign, 814
entropy

approximate, 194–196, 581, 587–588, 590
Kolmogorov, 581, 587–588
Shannon conditional, 581, 590
estimate
depth of anesthesia by the electroencephalogram, 581
invariant, 785–786, 788
rule, 106

F

feature
effective, 706
map, 953
morphological, 701, 709
Fisher's criterion, 581, 595, 598, 604–606
form coefficient, 558
Fourier series, 547
formula
Frenet, 572
Gauss quadrature, 557
function
autocorrelation, 215, 245, 753
cross-correlation, 216
decision, 105, 107, 581, 594, 599, 603, 606, 609
Haar, 546, 773
instrumental response, 745–746, 748, 752
local homogeneous smoothing, 776
local homogeneous restoring, 776
maximum plausibility, 108
point spread, 749
radial, 706
ridge, 180
Walsh, 548
window, 383
fundamental harmonic, 716

G

Gaussian profile, 711
GC-content, 551
generalized Catalan numbers, 412, 418, 424, 431
generating
rules, 16
descriptive tree, 4–5
geometric parameters of vessels, 710
Gibbs phenomenon, 550

H

hypertensive retinopathy, 700

I

Image
analysis, 123, 938
biomedical, 2, 10, 19
complexity, 122
comparable in shape, 108
component, 125
decomposition, 131
equivalence, 4
equivalent in shape, 88
fundus, 706
formalization space, 4, 7
hyperspectral, 312, 364
implementation, 14
low-count, 324
model, 2, 7, 14, 123
model construction, 16
mosaic, 91
processing, 124, 701, 703
raster terrain, 800
reconstruction, 123, 182
reducing to a recognizable form, 5–6, 11, 13, 25
representation, 7, 13
restoration, 123
shape, 87–88, 90
smoothing, 123
image algebra
descriptive, 1–9, 26–38
descriptive with one ring, 1–10, 33, 40–41
Ritter's standard, 34
indices of morphological dependency and connectedness, 104
infeasible system of constraints, 461

informational structure for generating descriptive algorithmic image recognition schemes, 7
inhomogeneous filter, 947
intellectual technology for graphic documents processing, 794
intraventricular blood flow, 730
inverse
  high-dimensional, 505
  laser spectroscopy, 517
  problem, 507, 560
isoline, 813
isomap, 369

J

joined phase unwrapping, 741

L

Laplacian eigenmaps, 365–369
Levenstein distance, 550
locally linear embedding, 369

M

machine learning, 325, 332, 341, 460, 485, 507, 803, 809, 836, 843, 853, 916, 981, 986
Manhattan metric, 294
mapping
  cardiac color Doppler flow, 730
  Hellinger divergence preserving, 380
  nonlinear, 367
  spectral angle preserving, 379
  spectral correlation preserving, 380
  spectral information divergence preserving, 379
mathematical theory of image analysis, 3, 5
mathematical theory of pattern recognition, 5
maximal feasible subsystem, 461
  Graph of m.f.s., 462
  Hypergraph of m.f.s., 462
  Criterion of m.f.s. hypergraph, 463
maximum likelihood principle, 331
medical diagnostic, 581, 700
  criteria, 702
  devices, 703
  features, 703, 705–706, 709
  information, 701
method
  adaptive threshold, 710
  collective, 489–503
  conjugation index, 295, 312–319
  correlation-extreme contour, 783
  global minimum location, 793
  iterative regularization, 729
  laser fluorescent microscopy, 747
  Mazurov committee, 27, 459–484
  object extraction nonlocal, 113
  parabolic, 793
  Rastrigin-Erenshtein collective decision rules, 27
  symmetry axes, 791
microcirculatory system of the eye, 715
micro vessel changes, 706
modalities of researcher's subjective judgements, 105
model
  absolute ignorance, 106
  algebraic, 66–68
  behavioral vision, 947
  blood vessel segment, 715
  construction, 16
  descriptive image, 4–5, 7, 9, 13, 34
  digital surface, 800
  digital terrain, 800
  exact knowledge, 104, 106
  formal image, 17
  Gaussian mixture, 342
  generating image, 18
  heuristic, 916
  parametric image, 20–22, 24
  procedural image, 18–19, 21
  smoothing parameters of acoustic, 987–989
  subjective, 106
  subjective morphological, 108–109
  visual information, 104

modeling
  implicit pronunciation variation, 976, 985
  explicit pronunciation variation, 976, 977
  mathematical formalism of subjective, 104
  unformalized, incomplete and unreliable knowledge, 104
  subjective, 106
molecular dynamics, 573
morphological dependency, connectedness, 104
morphological proximity, 94
moving and stationary target acquisition, 314
multidimensional integral, 403
multilayer construction, 915
multiple classifiers, 28
multiple testing, 692

N

navigation behavior, 944
neural network, 306
noise resilience, 516
noise/signal, 707
nonlinear
  components, 581
  dimensionality reduction, 365
  mapping, 367

O

object image extraction, 104
objectivity and constancy of perception, 942
oblique projection, 101–102
oblique projector, 101–102
observation selectivity, 946
OpenMP
ophthalmology monitoring programs, 700
optic nerve, 701
optimal estimate minimizing the plausibility of an error, 108
optimal search strategy, 433–434
optimum identification rules, 107
order statistics, 384

P

parametric optimization, 789
pathologic vessel change, 705
phase unwrapping, 730
photo activation localization microscopy, 749
photon-counting image sensors, 326
photon-counting sensor, 326, 328
piecewise linear committee classifier, 481
planar projection, 709
plausibility and belief
  distribution, 105
  measure, 104
  loss, 107
  scales of measures, 105
  transitional, 106
point-pulse source, 431, 438
Poisson point process, 327
Poisson source, 394, 433
polynomial
  Charlier, 549
  Chebyshev, 549
  Hermite, 550
  Hahn, 549
  Jacobi, 550
  Kravchuk, 549
  Laguerre, 550
  Legendre, 550
  Meixner, 549
post-traumatic epileptiform activity, 657–659
preclinical diagnostics
  Parkinson's Disease, 658
  Post-traumatic Epileptiform Activity, 658
primitive element, 620
priority queue, 375
problem
  ballot, 420, 427
  classification, 27, 497–503, 706

cluster analysis, 492–497
geophysics inverse, 527–529
high-dimensional inverse, 505
inverse, 507, 560
laser spectroscopy inverse, 517
minimal affine separating committee, 482
minimization, 734
morphological filtering, 104
multiclass, 605–606, 609
ophthalmological diagnostic automation, 66–68
optimum estimation, 108
recognition, 27–28, 39, 68
synthesizing pattern recognition algorithms, 915
projection operator onto shape of images, 89

R

recognition
atrial fibrillation, 586, 593–595
biomedical signals, 579–610
face, 314
moving and stationary target, 599
symbols on maps, 794
relative shapes as oblique projectors, 111
representation
combinatorial structures of local neighborhoods representation, 10, 17
hierarchical structures for data representation, 777
researcher's subjective judgements, 104–105
flow 2D restoration, 741
restoration of vessels planar structures, 709

S

shape
absolute, 104, 113
connectedness, 112
image, 87–88, 90
images comparable in shape, 108
images equivalent in shape, 88
mutually morphologically independent, 111
projection operator onto image shape, 89
relative, 104, 112
rival similarity of image shape, 97
subjective shape of the uncertain element, 109
Shlesinger two-dimensional grammars, 28
signal classification, 581, 585
similarity
DNA fragments, 550
numerical measure of rival similarity, 99
rival, 98
simultaneous determination, 515
source localization, 562
space imagery digitizing, 801
space partitioning, 372–376
spatial context, 381–382
spectral angle mapper, 378
spectral information divergence, 380
spectroscopy of single molecules, 748
spherical harmonics, 561
splines, 570
stimulated emission depletion, 749–750
stochastic gradient descent, 371
stochastic optical reconstruction microscopy, 749
structural motifs, 569
structural optimization, 793
structuring element, 16
subjective independence, 110
subjective judgements as uncertain propositions, 105
support subspaces, 297
support vector machine, 294
synthetic aperture radar, 314
system
AlluraXper visualization, 704

cardiac monitoring, 599, 606
computer system for geometrical analysis of blood vessels diagnostic images, 702
geoinformation, 800
vascular, 710

T

terrain object, 804
test
Indian Pines test, 319
permutation test, 685
texture, 180
theory
descriptive image analysis, 4
Grenander's pattern, 6, 27
Marr's computational vision, 6
three-dimensional structure, 707
time–frequency distribution, 657, 667
topologic information, 717
topological characteristics, 161
transform(ation)
generating, 16
Hough, 760
parametric, 15
procedural, 16
Radon, 180
two-dimensional variations, 122

V

valid partitioning, 678
vascular disease, 701, 710
vessel separation, 702
vessel tree, 707
viewing trajectories, 958
visual sensor, 946
visual uncertainty, 942

W

wavelet spectrogram, 661, 663
weak radiation registration, 325

X

X-ray angiographic projections, 709

Y

Young tables, 411

Z

zero-sum committee game, 467