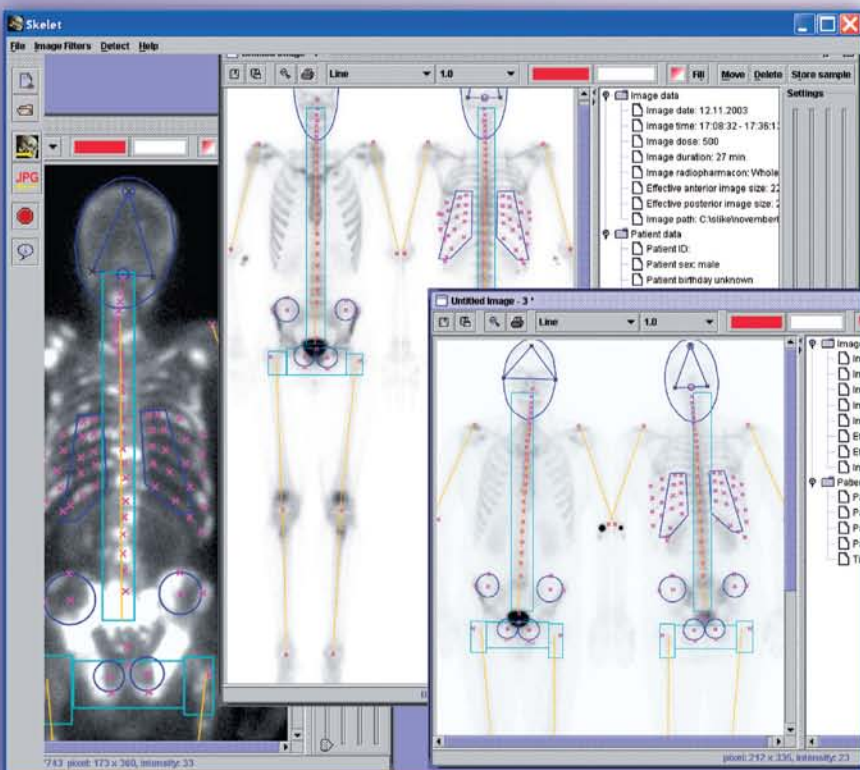
COMPUTATIONAL INTELLIGENCE IN MEDICAL IMAGING

Techniques and Applications



EDITED BY

Gerald Schaefer
Aboul Ella Hassanien
Jianmin Jiang

CRC Press
Taylor & Francis Group
A CHAPMAN & HALL BOOK

Computational Intelligence in Medical Imaging

Techniques and Applications

Computational Intelligence in Medical Imaging

Techniques and Applications

Edited by
Gerald Schaefer
Aboul Ella Hassanien
Jianmin Jiang

CRC Press
Taylor & Francis Group
Boca Raton London New York

CRC Press is an imprint of the
Taylor & Francis Group, an **informa** business
A CHAPMAN & HALL BOOK

Chapman & Hall/CRC
Taylor & Francis Group
6000 Broken Sound Parkway NW, Suite 300
Boca Raton, FL 33487-2742

Printed in the United States of America on acid-free paper
10 9 8 7 6 5 4 3 2 1

International Standard Book Number-13: 978-1-4200-6059-1 (Hardcover)

Library of Congress Cataloging-in-Publication Data

Computational intelligence in medical imaging techniques and applications /
editors, Gerald Schaefer, Aboul Ella Hassanien, and Jianmin Jiang.
p. ; cm.
Includes bibliographical references and index.
ISBN 978-1-4200-6059-1 (alk. paper)
1. Diagnostic imaging--Data processing. 2. Computational intelligence. I. Schaefer, Gerald. II. Hassanien, Aboul Ella. III. Jiang, J., Ph. D. IV. Title.
[DNLM: 1. Diagnosis, Computer-Assisted--methods. 2. Diagnosis, Computer-Assisted--trends. WB 141 C738 2008]

RC78.7.D53C656 2008
616.07'540285--dc22 2008040413

Visit the Taylor & Francis Web site at
http://www.taylorandfrancis.com

and the CRC Press Web site at
http://www.crcpress.com

Contents

Preface **vii**

Editors **ix**

Contributors **xi**

1 Computational Intelligence on Medical Imaging with Artificial Neural Networks **1**
Z. Q. Wu, Jianmin Jiang, and Y. H. Peng

2 Evolutionary Computing and Its Use in Medical Imaging **27**
Lars Nolle and Gerald Schaefer

3 Rough Sets in Medical Imaging: Foundations and Trends **47**
Aboul Ella Hassanien, Ajith Abraham, James F. Peters, and Janusz Kacprzyk

4 Early Detection of Wound Inflammation by Color Analysis **89**
Peter Plassmann and Brahima Belem

5 Analysis and Applications of Neural Networks for Skin Lesion Border Detection **113**
Maher I. Rajab

6 Prostate Cancer Classification Using Multispectral Imagery and Metaheuristics **139**
Muhammad Atif Tahir, Ahmed Bouridane, and Muhammad Ali Roula

7 Intuitionistic Fuzzy Processing of Mammographic Images **167**
Ioannis K. Vlachos and George D. Sergiadis

8 Fuzzy C-Means and Its Applications in Medical Imaging **213**
Huiyu Zhou

9 Image Informatics for Clinical and Preclinical Biomedical Analysis **239**
Kenneth W. Tobin, Edward Chaum, Jens Gregor, Thomas P. Karnowski, Jeffery R. Price, and Jonathan Wall

10 Parts-Based Appearance Modeling of Medical Imagery **291**
Matthew Toews and Tal Arbel

11 Reinforced Medical Image Segmentation **327**
Farhang Sahba, Hamid R. Tizhoosh, and Magdy M. A. Salama

12 Image Segmentation and Parameterization for Automatic Diagnostics of Whole-Body Scintigrams: Basic Concepts **347**
Luka Šajn and Igor Kononenko

13 Distributed 3-D Medical Image Registration Using Intelligent Agents **379**
Roger J. Tait, Gerald Schaefer, and Adrian A. Hopgood

14 Monte Carlo–Based Image Reconstruction in Emission Tomography **407**
Steven Staelens and Ignace Lemahieu

15 Deformable Organisms: An Artificial Life Framework for Automated Medical Image Analysis **433**
Ghassan Hamarneh, Chris McIntosh, Tim McInerney, and Demetri Terzopoulos

Index **475**

Preface

Medical imaging is an indispensible tool for many branches of medicine. It enables and facilitates the capture, transmission, and analysis of medical images and aids in medical diagnoses. The use of medical imaging is still on the rise with new imaging modalities being developed and continuous improvements being made to devices' capabilities. Recently, computational intelligence techniques have been employed in various applications of medical imaging and have been shown to be advantageous compared to classical approaches, particularly when classical solutions are difficult or impossible to formulate and analyze. In this book, we present some of the latest trends and developments in the field of computational intelligence in medical imaging.

The first three chapters present the current state of the art of various areas of computational intelligence applied to medical imaging. Chapter 1 details neural networks, Chapter 2 reviews evolutionary optimization techniques, and Chapter 3 covers in detail rough sets and their applications in medical image processing.

Chapter 4 explains how neural networks and support vector machines can be utilized to classify wound images and arrive at decisions that are comparable to or even more consistent than those of clinical practitioners. Neural networks are also explored in Chapter 5 in the context of accurately extracting the boundaries of skin lesions, a crucial stage for the identification of melanoma. Chapter 6 discusses tabu search, an intelligent optimization technique, for feature selection and classification in the context of prostate cancer analysis.

In Chapter 7, the authors demonstrate how image processing techniques based on intuitionistic fuzzy sets can successfully handle the inherent uncertainties present in mammographic images. Fuzzy logic is also employed in Chapter 8, where fuzzy set–based clustering techniques for medical image segmentation are discussed.

A comprehensive system for handling and utilizing biomedical image databases is described in Chapter 9: The features extracted from medical images are encoded within a Bayesian probabilistic framework that enables learning from previously retrieved relevant images. Chapter 10 explores how machine learning techniques are used to develop a statistical parts-based appearance model that can be used to encapsulate the natural intersubject anatomical variance in medical images.

In Chapter 11, a multistage image segmentation algorithm based on reinforcement learning is introduced and successfully applied to the problem of prostate segmentation in transrectal ultrasound images. Chapter 12 presents a machine learning approach for automatic segmentation and diagnosis of bone scintigraphy. Chapter 13 employs a set of intelligent agents that communicate via a blackboard architecture to provide accurate and efficient 3-D medical image segmentation.

Chapter 14 explains how Monte Carlo simulations are employed to perform reconstruction of SPECT and PET tomographic images. Chapter 15 discusses the use of artificial life concepts to develop intelligent, deformable models that segment and analyze structures in medical images.

Obviously, a book of 15 chapters is nowhere near sufficient to encompass all the exciting research that is being conducted in utilizing computational intelligence techniques in the context of medical imaging. Nevertheless, we believe the chapters that were selected from among almost 40 proposals and rigorously reviewed by three experts present a good snapshot of the field. This work will prove useful not only in documenting recent advances but also in stimulating further research in this area.

Gerald Schaefer, Aboul Ella Hassanien, Jianmin Jiang

We are grateful to the following reviewers:

Rocio Alba-Flores
Lucia Ballerini
Hueseyin Cakmak
Marina Chukalina
Sergio Damas
Sabine Dippel
Christian Dold
Emanuele Frontoni
Peter Goebel
Yu-Le Huang
Jiri Jan
Jean-Claude Klein
Syoji Kobashi
Francesco Masulli
Kalana Mendis
Roberto Morales
Henning Mueller
Mike Nachtegael
Tomoharu Nakashima
Dmitry Nikolaev
Lars Nolle
Peter Plassmann
Stefan Schulte
George Sergiadis
Stephen Smith
Roman Starosolski
Kenji Suzuki
Gui Yun Tian
Michal Zavisek
Primo Zingaretti
Reyer Zwiggelaar

Editors

Gerald Schaefer obtained his BSc in computing from the University of Derby and his PhD in computer vision from the University of East Anglia. He worked as a research associate (1997–1999) at the Colour & Imaging Institute, University of Derby, as a senior research fellow at the School of Information Systems, University of East Anglia (2000–2001), and as a senior lecturer in computing at the School of Computing and Informatics at Nottingham Trent University (2001–2006). In September 2006, he joined the School of Engineering and Applied Science at Aston University. His research interests include color image analysis, image retrieval, medical imaging, and computational intelligence. He is the author of more than 150 scientific publications in these areas.

Aboul Ella Hassanien is an associate professor in the Computer and Information Technology Department at Cairo University. He works in a multidisciplinary environment involving computational intelligence, information security, medical image processing, data mining, and visualization applied to various real-world problems. He received his PhD in computer science from Tokyo Institute of Technology, Japan. He serves on the editorial board of several reputed international journals, has guest edited many special issues on various topics, and is involved in the organization of several conferences. His research interests include rough set theory, wavelet theory, medical image analysis, fuzzy image processing, information security, and multimedia data mining.

Jianmin Jiang is a full professor of digital media at the School of Informatics, University of Bradford, United Kingdom. He received his BSc from Shandong Mining Institute, China in 1982; his MSc from China University of Mining and Technology in 1984; and his PhD from the University of Nottingham, United Kingdom in 1994. His research interests include image/video processing in compressed domains, medical imaging, machine learning and AI applications in digital media processing, retrieval, and analysis. He has published more than 200 refereed research papers and is the author of one European patent (EP01306129) filed by British Telecom Research Lab. He is a chartered engineer, a fellow of IEE, a fellow of RSA, a member of EPSRC College, an EU FP-6/7 proposal evaluator, and a consulting professor at the Chinese Academy of Sciences and Southwest University, China.

Contributors

Ajith Abraham
Center for Quantifiable Quality of Service in Communication Systems
Norwegian University of Science and Technology
Trondheim, Norway

Tal Arbel
Centre for Intelligent Machines
McGill University
Montreal, Quebec, Canada

Brahima Belem
Department of Computing and Mathematical Sciences
University of Glamorgan
Pontypridd, Wales, United Kingdom

Ahmed Bouridane
Institute of Electronics, Communications and Information Technology (ECIT)
Queen's University Belfast
Belfast, Ireland

Edward Chaum
University of Tennessee Health Science Center
Hamilton Eye Institute
Memphis, Tennessee

Jens Gregor
Department of Computer Science
University of Tennessee
Knoxville, Tennessee

Ghassan Hamarneh
School of Computing Science
Simon Fraser University
Burnaby, British Columbia, Canada

Aboul Ella Hassanien
Information Technology Department
Cairo University
Giza, Egypt

Adrian A. Hopgood
Faculty of Computing Sciences and Engineering
De Montfort University
Leicester, United Kingdom

Jianmin Jiang
School of Informatics
University of Bradford
Bradford, West Yorkshire, United Kingdom

Janusz Kacprzyk
Systems Research Institute
Polish Academy of Sciences
Warsaw, Poland

Thomas P. Karnowski
Image Science and Machine Vision Group
Oak Ridge National Laboratory
Oak Ridge, Tennessee

Igor Kononenko
Faculty of Computer and Information Science
University of Ljubljana
Ljubljana, Slovenia

Ignace Lemahieu
University of Ghent – DOZA
Ghent, Belgium

Tim McInerney
Department of Computer Science
Ryerson University
Toronto, Ontario, Canada

Chris McIntosh
School of Computing Science
Simon Fraser University
Burnaby, British Columbia, Canada

Lars Nolle
School of Science and Technology
Nottingham Trent University
Nottingham, United Kingdom

Y. H. Peng
School of Informatics
University of Bradford
Bradford, West Yorkshire, United Kingdom

James F. Peters
Department of Electrical and Computer Engineering
University of Manitoba
Winnipeg, Manitoba, Canada

Peter Plassmann
Department of Computing and Mathematical Sciences
University of Glamorgan
Pontypridd, Wales, United Kingdom

Jeffery R. Price
Image Science and Machine Vision Group
Oak Ridge National Laboratory
Oak Ridge, Tennessee

Maher I. Rajab
Computer Engineering Department
College of Computer and Information Systems
Umm Al-Qura University
Mecca, Saudi Arabia

Mohammad Ali Roula
Department of Electronics and Computer Systems Engineering
University of Glamorgan
Pontypridd, Wales, United Kingdom

Farhang Sahba
Department of Systems Design Engineering
University of Waterloo
Waterloo, Ontario, Canada

Luka Šajn
Faculty of Computer and Information Science
University of Ljubljana
Ljubljana, Slovenia

Magdy M. A. Salama
Department of Electrical and Computer Engineering
University of Waterloo
Waterloo, Ontario, Canada

Gerald Schaefer
School of Engineering and Applied Science
University of Birmingham
Birmingham, United Kingdom

George D. Sergiadis
Department of Electrical and Computer Engineering
Aristotle University of Thessaloniki
Thessaloniki, Greece

Steven Staelens
IBITECH-MEDISIP
Ghent University – IBBT
Ghent, Belgium

Muhammad Atif Tahir
Faculty of CEMS
University of the West of England
Bristol, United Kingdom

Roger J. Tait
School of Computing and Informatics
Nottingham Trent University
Nottingham, United Kingdom

Demetri Terzopoulos
Department of Computer Science
University of California
Los Angeles, California

Hamid R. Tizhoosh
Department of Systems Design Engineering
University of Waterloo
Waterloo, Ontario, Canada

Kenneth W. Tobin
Image Science and Machine Vision Group
Oak Ridge National Laboratory
Oak Ridge, Tennessee

Matthew Toews
Centre for Intelligent Machines
McGill University
Montreal, Quebec, Canada

Ioannis K. Vlachos
Department of Electrical and Computer Engineering
Aristotle University of Thessaloniki
Thessaloniki, Greece

Jonathan Wall
Amyloid and Preclinical and Diagnostic Molecular Imaging Laboratory
University of Tennessee Graduate School of Medicine
Knoxville, Tennessee

Z. Q. Wu
School of Informatics
University of Bradford
Bradford, West Yorkshire, United Kingdom

Huiyu Zhou
Department of Electronic and Computer Engineering
School of Engineering and Design
Brunel University
Uxbridge, Middlesex, United Kingdom

Chapter 1

Computational Intelligence on Medical Imaging with Artificial Neural Networks

Z. Q. Wu, Jianmin Jiang, and Y. H. Peng

Contents

1.1 Introduction 2
1.2 Neural Network Basics 3
1.3 Computer-Aided Diagnosis (CAD) with Neural Networks 5
1.4 Medical Image Segmentation and Edge Detection with Neural Networks 10
1.5 Medical Image Registration with Neural Networks 13
1.6 Other Applications with Neural Networks 16
1.7 Conclusions 19
Acknowledgment 21
References 22

Neural networks have been widely reported in the research community of medical imaging. In this chapter, we provide a focused literature survey on neural network development in computer-aided diagnosis (CAD), medical image segmentation and edge detection toward visual content analysis, and medical image registration for its preprocessing and postprocessing. From among all these techniques and algorithms, we select a few representative ones to provide inspiring examples to illustrate (a) how a known neural network with fixed structure and training procedure can be applied to resolve a medical imaging problem; (b) how medical images can be analyzed, processed, and characterized by neural networks; and (c) how neural networks can be expanded further to resolve problems relevant to medical imaging. In the concluding section, a comparison of all neural networks is included to provide a global view on computational intelligence with neural networks in medical imaging.

1.1 Introduction

An artificial neural network (ANN) is an information processing system that is inspired by the way biological nervous systems store and process information like human brains. It contains a large number of highly interconnected processing neurons working together in a distributed manner to learn from the input information, to coordinate internal processing, and to optimize its final output. In the past decades, neural networks have been successfully applied to a wide range of areas, including computer science, engineering, theoretical modeling, and information systems. Medical imaging is another fruitful area for neural networks to play crucial roles in resolving problems and providing solutions. Numerous algorithms have been reported in the literature applying neural networks to medical image analysis, and we provide a focused survey on computational intelligence with neural networks in terms of (a) CAD with specific coverage of image analysis in cancer screening, (b) segmentation and edge detection for medical image content analysis, (c) medical image registration, and (d) other applications covering medical image compression, providing a global view on the variety of neural network applications and their potential for further research and developments.

Neural network applications in CAD represent the mainstream of computational intelligence in medical imaging. Their penetration and involvement are comprehensive for almost all medical problems because (a) neural networks can adaptively learn from input information and upgrade themselves in accordance with the variety and change of input content; (b) neural networks can optimize the relationship between the inputs and outputs via distributed computing, training, and processing, leading to reliable solutions desired by specifications; (c) medical diagnosis relies on visual inspection, and medical imaging provides the most important tool for facilitating such inspection and visualization.

Medical image segmentation and edge detection remains a common problem fundamental to all medical imaging applications. Any content analysis and regional inspection requires segmentation of featured areas, which can be implemented via edge detection and other techniques. Conventional approaches are typified by a range of well-researched algorithms, including watershed, region-growing, snake modeling, and contour detection. In comparison, neural network approaches exploit the learning capability and training mechanism to classify medical images into content-consistent regions to complete segmentations as well as edge detections.

Another fundamental technique for medical imaging is registration, which plays important roles in many areas of medical applications. Typical examples include wound care, disease prediction, and health care surveillance and monitoring. Neural networks can be designed to provide alternative solutions via competitive learning, self-organizing, and clustering to process input features and find the best possible alignment between different images or data sets.

The remainder of this chapter provides useful insights for neural network applications in medical imaging and computational intelligence. We explain the basics of neural networks to enable beginners to understand the structure, connections, and neuron functionalities. Then we present detailed descriptions of neural network applications in CAD, image segmentation and edge detection, image registration, and other areas.

1.2 Neural Network Basics

To enable understanding of neural network fundamentals, to facilitate possible repetition of those neural networks introduced and successfully applied in medical imaging, and to inspire further development of neural networks, we cover essential basics in this section about neural networks to pave the way for the rest of the chapter in surveying neural networks. We start from a theoretical model of one single neuron and then introduce a range of different types of neural networks to reveal their structure, training mechanism, operation, and functions.

The basic structure of a neuron can be theoretically modeled as shown in Figure 1.1.

Figure 1.1 shows the model of a single neuron, where $X\{x_i, i = 1, 2, \ldots, n\}$ represents the inputs to the neuron and Y represents the output. Each input is multiplied by its weight w_i, a bias b is associated with each neuron, and their sum goes through a transfer function f. As a result, the relationship between input and output can be described as follows.

$$Y = f\left(\sum_{i=1}^{n} w_i x_i + b\right) \tag{1.1}$$

A range of transfer functions have been developed to process the weighted and biased inputs. Four of the basic transfer functions widely adopted for medical image processing are illustrated in Figure 1.2.

Via selection of transfer function and connection of neurons, various neural networks can be constructed to be trained for producing the specified outputs. Major neural networks commonly used for medical image processing are

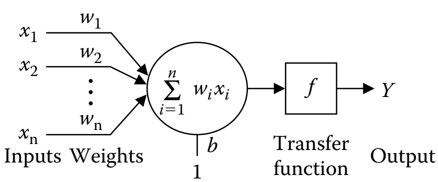


FIGURE 1.1: The model of a neuron.

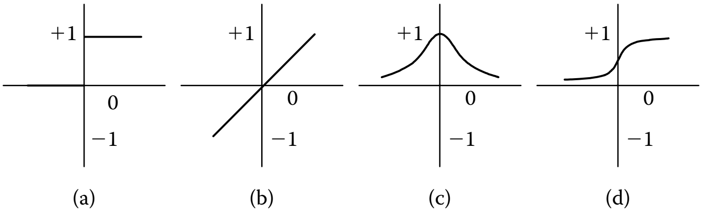


FIGURE 1.2: Four widely adopted transfer functions: (a) hardlimit, (b) linear, (c) RBF, and (d) sigmoid.

classified as feedforward neural network, feedback network, and self-organizing map. The learning paradigms for the neural networks in medical image processing generally include supervised networks and unsupervised networks. In supervised training, the training data set consists of many pairs in the source and target patterns. The network processes the source inputs and compares the resulting outputs against the target outputs, and adjusts its weights to improve the correct rate of the resulting outputs. In unsupervised networks, the training data set does not include any target information.

A general feedforward network [1] often consists of multiple layers, typically including one input layer, a number of hidden layers, and an output layer. In the feedforward neural networks, the neurons in each layer are only fully interconnected with the neurons in the next layer, which means signals or information being processed travel along a single direction.

A back-propagation (BP) network [2] is a supervised feedforward neural network, and it is a simple stochastic gradient descent method to minimize the total squared error of the output computed by the neural network. Its errors propagate backwards from the output neurons to the inner neurons. The processes of adjusting the set of weights between the layers and recalculating the output continue until a stopping criterion is satisfied.

The radial basis function (RBF) [3] network is a three-layer, supervised feedforward network that uses a nonlinear transfer function (normally the Gaussian) for the hidden neurons and a linear transfer function for the output neurons. The Gaussian is applied to the net input to produce a radial function of the distance between each pattern vector and each hidden unit weight vector.

The feedback (or recurrent) neural network [4] can have signals traveling in both directions by introducing loops. Their state is changing continuously until they reach an equilibrium point. They remain at the equilibrium point until the input changes and a new equilibrium must be found. They are powerful but can get extremely complicated.

The Hopfield network [4] is a typical feedback, and its inspiration is to store certain patterns in a manner similar to the way the human brain stores memories. The Hopfield network has no special input or output neurons, but all neurons are both input and output, and all of them connect to all others in both directions. After receiving the input simultaneously by all the neurons, they output to each other, and the process does not stop until a stable state is reached. In the Hopfield network, it is simple to set up the weights between neurons in order to set up a desired set of patterns as stable class patterns. The Hopfield network is an unsupervised learning network and thus does not require a formal training phase.

Quite different from feedforward and feedback networks, the Kohonen neural network (a self-organizing map, SOM) [5] learns to classify input vectors according to how they are grouped in the input space. In the network, a set of artificial neurons learns to map points in an input space to the coordinates in an output space. Each neuron stores a weight vector (an array of weights), each of which corresponds to one of the inputs in the data. When presented with a new input pattern, the neuron whose weight is closest in Euclidian space to the new input pattern is allowed to adjust its weight so that it gets closer to the input pattern. The Kohonen neural network uses a competitive learning algorithm to train itself in an unsupervised manner.

In Kohonen neural networks, each neuron is fed by input vector (data point) $x \in R^n$ through a weight vector $w \in R^n$. Each time a data point is input to the network, only the neuron j whose weight vector most resembles the input vector is selected to fire, according to the following rule:

$$j = \arg\min(\|x - w\|^2) \tag{1.2}$$

The firing or winning neuron j and its neighboring neurons i have their weight vectors w modified according to the following rule:

$$w_i(t+1) = w_i(t) + h_{ij}(\|r_i - r_j\|, t) \cdot (x(t) - w_i(t)) \tag{1.3}$$

where $h_{ij}(||r_i - r_j||, t)$ is a kernel defined on the neural network space as a function of the distance $||r_i - r_j||$ between the firing neuron j and its neighboring neurons i, and the time t defines the number of iterations. Its neighboring neurons modify their weight vectors so they also resemble the input signal, but less strongly, depending on their distance from the winner.

The remainder of the chapter provides detailed descriptions of computational intelligence in medical imaging with neural networks. Their recent applications are classified into four categories: CAD, image segmentation, registration, and other applications. Each section gives more details on an application in one of these categories and provides overviews of the other relevant applications. A comparison of neural networks is presented in Section 1.7.

1.3 Computer-Aided Diagnosis (CAD) with Neural Networks

Neural networks have been incorporated into many CAD systems, most of which distinguish cancerous signs from normal tissues. Generally, these systems enhance the images first and then extract interesting regions from the images. The values of many features are calculated based on the extracted regions and are forwarded to neural works that make decisions in terms of learning, training, and optimizations. Among all applications, early diagnosis of breast cancers and lung cancers represents the most typical examples in the developed CAD systems.

Ge and others [6] developed a CAD system to identify microcalcification clusters automatically on full-field digital mammograms. The main procedures of the CAD system included six stages: preprocessing, image enhancement, segmentation of microcalcification candidates, false positive (FP) reduction for individual microcalcifications, regional clustering, and FP reduction for clustered microcalcifications.

To reduce FP individual microcalcifications, a convolution neural network (CNN) was employed to analyze 16×16 regions of interest centered at the candidate derived from segmentations. The CNN was designed to simulate the vision of vertebrate animals and could be considered a simplified vision machine designed to perform the classification of the regions into two output types: disease and nondisease. Their CNN contained an input layer with 14 neurons, two hidden layers with 10 neurons each, and one output layer. The convolution kernel sizes of the first group of filters between the input and the first hidden layer were designed as 5×5, and those of the second group of filters between the first and second hidden layers were 7×7. The images in each layer were convolved with convolution kernels to obtain the pixel values to be transferred to the following layer. The logistic sigmoid function was chosen as the transfer function for both the hidden neurons and output neurons. An illustration of the neural network structure and its internal connections between the input layer, hidden layer, and output layers is given in Figure 1.3.

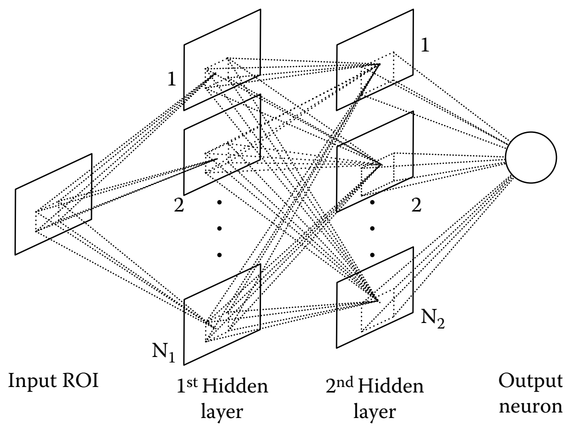


FIGURE 1.3: Schematic diagram of a CNN.

The convolution kernels are arranged in a way to emphasize a number of image characteristics rather than those less correlated values derived from feature spaces of input. These characteristics include (a) the horizontal versus vertical information, (b) local versus nonlocal information, and (c) image processing (filtering) versus signal propagation [7].

The CNN was trained using a backpropagation learning rule with the sum-of-squares error (SSE) function, which allowed a probabilistic interpretation of the CNN output, that is, the probability of correctly classifying the input sample as a true microcalcification region of interest (ROI).

At the stage of FP reduction for clustered microcalcifications, morphological features (such as the size, mean density, eccentricity, moment ratio, axis ratio features, and number of microcalcifications in a cluster) and features derived from the CNN outputs (such as the minimum, maximum, and mean of the CNN output values) were extracted from each cluster. For each cluster, 25 features (21 morphological plus 4 CNN features) were extracted. A linear discriminating analysis (LDA) classifier was then applied to differentiate clustered microcalcifications from FPs. The stepwise LDA feature selection involved the selection of three parameters for selection.

In the study by Ge and colleagues, a set of 96 images was split into a training set and a validation set, each with 48 images. An appropriate set of parameters was selected by searching in the parameter space for the combination of three parameters of the LDA that could achieve the highest classification accuracy with a relatively small number of features in the validation set. Then the three parameters of LDA were applied to select a final set of features and the LDA coefficients by using the entire set of 96 training images, which contained 96 true positive (TP) and over 500 FP clusters. The trained classifier was applied to a test subset to reduce the FPs in the CAD system [6].

To develop a computerized scheme for the detection of clustered microcalcifications in mammograms, Nagel and others [8] examined three methods of feature analysis: rule based (the method currently used), an ANN, and a combined method. The ANN method used a three-layer error-backpropagation network with five input units corresponding to the radiographic features of each microcalcification and one output unit corresponding to the likelihood of being a microcalcification. The reported work revealed that two hidden units were insufficient for good performance of the ANN, and it was necessary to have at least three hidden units to achieve adequate performance. However, the performance was not improved any further when the number of hidden units was increased over three. Therefore, the finalized ANN had five inputs, three hidden units, and one output unit. It was reported that such a combined method performed better than any method alone.

Papadopoulossa, Fotiadisb, and Likasb [9] presented a hybrid intelligent system for the identification of microcalcification clusters in digital mammograms, which could be summarized in three steps: (a) preprocessing and segmentation, (b) ROI specification, and (c) feature extraction and classification. In the classification schema, 22 features were automatically computed that referred either to individual microcalcifications or to groups of them. The reduction of FP cases was performed using an intelligent system containing two subsystems: a rule-based system and a neural network-based system. The rule construction procedure consisted of the feature identification step as well as the selection of the particular threshold value for each feature. Before using the neural network, the reduction in the number of features was achieved through principal component analysis (PCA), which transforms each 22-dimensional feature vector into a 9-dimensional feature vector as the input to the neural network. The neural network used for ROI characterization was a feedforward neural network with sigmoid hidden neuron (multilayer perceptron, MLP).

Christoyiani, Dermatas, and Kokkinakis [10] presented a method for fast detection of circumscribed mass in mammograms employing an RBF neural network (RBFNN). In the method, each neuron output was a nonlinear transformation of a distance measure of the neuron weights and its input vector. The nonlinear operator of the RBFNN hidden layer was implemented using a Cauchy-like probability density function. The implementation of RBFNN could be achieved by using supervised or unsupervised learning algorithms for an accurate estimation of the hidden layer weights. The k-means unsupervised algorithm was adopted to estimate the hidden-layer weights from a set of training data containing statistical features from both circumscribed lesions and normal tissue. After the initial training and the estimation of the hidden-layer weights, the weights in the output layer were computed by using Wincer-filter theory, or minimizing the mean square error (MSE) between the actual and the desired filter output.

Patrocinio and others [11] demonstrated that only several features, such as irregularity, number of microcalcifications in a cluster, and cluster area, were needed as the inputs of a neural network to separate images into two distinct classes: suspicious and probably benign. Setiono [12] developed an algorithm by pruning a feedforward neural network, which produced high accuracy rates for breast cancer diagnosis with a small number of connections. The algorithm extracted rules from a pruned network by considering only a finite number of hidden-unit activation values. Connections in the network were allowed only between input units and hidden units and between hidden units and output units. The algorithm found and eliminated as many unnecessary network connections as possible during the training process. The accuracy of the extracted rules from the pruned network is almost as high as the accuracy of the original network.

The abovementioned applications cover different aspects of applying neural networks, such as the number of neurons in the hidden layer, the reduction of features in classifications, and the reduction of connections for better efficiency. Similar improvements could be made in applying ANN to other practical utilizations rather than just in identifying microcalcification clusters.

ANN also plays an important role in detecting the cancerous signs in lungs. Xu and colleagues [13] developed an improved CAD scheme for the automated detection of lung nodules in digital chest images to assist radiologists who may miss up to 30% of the actually positive cases in their daily practice. In the CAD scheme, nodule candidates were selected initially by multiple gray-level thresholds of the difference image (subtraction of a signal-enhanced image and a signal-suppressed image) and then classified into six groups. A large number of FPs were eliminated by adaptive rule-based tests and an ANN.

Zhou and others [14] proposed an automatic pathological diagnosis procedure called neural ensemble-based detection that utilized an ANN ensemble to identify lung cancer cells in the specimen images of needle biopsies obtained from the bodies of the patients to be diagnosed. An ANN ensemble formed a learning paradigm while several ANNs were jointly used to solve a problem. The ensemble was built on a two-level ensemble architecture, and the predictions of those individual networks were combined by plurality voting.

Keserci and Yoshida [15] developed a CAD scheme for automated detection of lung nodules in digital chest radiographs based on a combination of morphological features and the wavelet snake. In their scheme, an ANN was used to efficiently reduce FPs by using the combined features. The scheme was applied to a publicly available database of digital chest images for pulmonary nodules. Qian and others [16] trained a computer-aided cytologic diagnosis (CACD) system to recognize expression of the cancer biomarkers histone H2AX in lung cancer cells and then tested the accuracy of this system to distinguish resected lung cancer from preneoplastic and normal tissues. The major characteristics of CACD algorithms were to adapt detection parameters according to cellular image contents. Coppini and colleagues [17] described a neural network–based system for the computer-aided detection of lung nodules in chest radiograms. The approach was based on multiscale processing and feedforward neural networks that allowed an efficient use of a priori knowledge about the shape of nodules and the background structure.

Apart from the applications in breast cancer and lung cancer, ANN has been adopted in many other analyses and diagnosis. Mohamed and others [18] compared bone mineral density (BMD) values for healthy persons and identified those with conditions known to be associated with BMD obtained from dual X-ray absorptiometry (DXA). An ANN was designed to quantitatively estimate site-specific BMD values in comparison with reference values obtained by DXA. Anthropometric measurements (i.e., sex, age, weight, height, body mass index, waist-to-hip ratio, and the sum of four skinfold thicknesses) were fed to an ANN as input variables. The estimates based on four input variables were generated as output and were generally identical to the reference values among all studied groups.

Scott [19] tried determining whether a computer-based scan analysis could assist clinical interpretation in this diagnostically difficult population. An ANN was created using only objective image-derived inputs to diagnose the presence of pulmonary embolism. The ANN predictions performed comparably to clinical scan interpretations and angiography results.

In all the applications mentioned above, the roles of ANNs have a common principle in the sense that most of them are applied to reduce FP detections in both mammograms and chest images via examining the features extracted from the suspicious regions. As a matter of fact, ANN is not limited to academic research but also plays important roles in commercially available diagnosis systems, such as ImageChecker for mammograms.

1.4 Medical Image Segmentation and Edge Detection with Neural Networks

Medical image segmentation is a process for dividing a given image into meaningful regions with homogeneous properties. Image segmentation is an indispensable process in outlining boundaries of organs and tumors and in the visualization of human tissues during clinical analysis. Therefore, segmentation of medical images is very important for clinical research, diagnosis, and applications, leading to requirement of robust, reliable, and adaptive segmentation techniques.

Kobashi and others [20] proposed an automated method to segment the blood vessels from three-dimensional (3-D) time-of-flight magnetic resonance angiogram (MRA) volume data. The method consisted of three steps: removal of the background, volume quantization, and classification of primitives by using an artificial neural network.

After volume quantization by using a watershed segmentation algorithm, the primitives in the MRA image stand out. To further improve the result of segmentation, the obtained primitives had to be separated into the blood vessel class and the fat class. Three features and a three-layered, feedforward neural network were adopted for the classification. Compared with the fat, the blood vessel is like a tube—long and narrow. Two features, vascularity and narrowness, were introduced to measure such properties. Because the histogram of blood vessels is quite different from that of the fat in shapes, the third feature, histogram consistency, was added for further improvement of the segmentation.

The feedforward neural network is composed of three layers: an input layer, a hidden layer, and an output layer. The structure of the described neural network is illustrated in Figure 1.4.

As seen, three input units were included at the input layer, which was decided by the number of features extracted from medical images. The number of neurons in the output layer was one to produce two classes. The number of neurons in the hidden layer was usually decided by experiments. Generally, a range of different numbers were tried in the hidden layer, and the number that achieved the best training results was selected.

In the proposed method, the ANN classified each primitive, which was a clump of voxels, by evaluating the intensity and the 3-D shape. In their

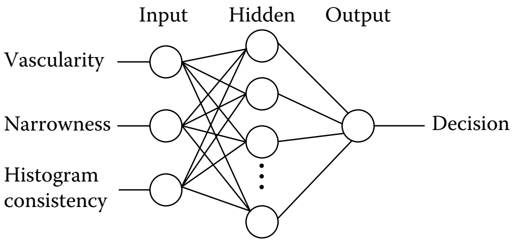


FIGURE 1.4: Three-layer feedforward neural network.

experiments, the ANN was trained using 60 teaching data sets derived from an MRA data set. Each primitive was classified into the blood vessel (indicated by the value of 1) or the fat (indicated by the value of 0), and the values of the three features were calculated. All these values were fed into the feedforward ANN for training the weights of the neurons. Seven new MRA data, whose primitives were unclassified, were fed into the trained neural network for testing. The segmentation performance was measured by the value of accuracy, as defined in Equation 1.4, and the rate achieved by the reported algorithm is 80.8% [20].

$$Accuracy = \frac{Number\ of\ correctly\ classified\ primitives}{Total\ number\ of\ primitives} \times 100\% \tag{1.4}$$

Apart from the work proposed by Kobashi and colleagues in ANN-based segmentation, there are many applications for the images generated by computed tomography (CT) and magnetic resonance imaging (MRI). Middleton and Damper [21] combined use of a neural network (an MLP, a type of feedforward neural network) and active contour model ("snake") to segment structures in magnetic resonance (MR) images. The highlights of the reported work can be summarized by the following two steps:

1. The perceptron was trained to produce a binary classification of each pixel as either a boundary or a nonboundary;
2. The resulting binary (edge-point) image formed the external energy function for a snake model, which was applied to link the candidate boundary points into a continuous and closed contour.

Lin [22] applied the Hopfield neural network (a feedback neural network) with penalized fuzzy c-means (FCM) technique to medical image segmentation. In the algorithm, the pixels with their first- and second-order moments constructed from their n nearest neighbors as a training vector were mapped to a two-dimensional (2-D) Hopfield neural network for the purpose of classifying the image into suitable regions.

Lin and colleagues [23] generalized the Kohonen competitive learning (KCL) algorithm with fuzzy and fuzzy-soft types called fuzzy KCL (FKCL) and fuzzy-soft KCL (FSKCL). These KCL algorithms fused the competitive learning with soft competition and FCM membership functions. These generalized KCLs were applied to MRI and MRA ophthalmological segmentations. It was found that these KCL-based MRI segmentation techniques were useful in reducing medical image noise effects using a learning mechanism. The FSKCL algorithm was recommended for use in MR image segmentation as an aid to small lesion diagnosis.

Dokur and Olmez [24] proposed a quantizer neural network (QNN) for the segmentation of MR and CT images. QNN was a novel neural network structure and was trained by genetic algorithms. It was comparatively examined with an MLP and a Kohonen network for the segmentation of MR and CT head images. They reported that QNN achieved the best classification performance with fewer neurons after a short training time.

Stalidis and others [25] presented an integrated model-based processing scheme for cardiac MRI, which was embedded in an interactive computing environment suitable for quantitative cardiac analysis. The scheme provided a set of functions for the extraction, modeling, and visualization of cardiac shape and deformation. In the scheme, a learning segmentation process incorporating a generating–shrinking neural network was combined with a spatiotemporal parametric model through functional basis decomposition.

Chang and Ching [26] developed an approach for medical image segmentation using a fuzzy Hopfield neural network based on both global and local gray-level information. The membership function simulated with neuron outputs was determined using a fuzzy set, and the synaptic connection weights between the neurons were predetermined and fixed in order to improve the efficiency of the neural network.

Shen and others [27] proposed a segmentation technique based on an extension to the traditional FCM clustering algorithm. In their work, a neighborhood attraction, which was dependent on the relative location and features of neighboring pixels, was shown to improve the segmentation performance, and the degree of attraction was optimized by a neural-network model. Simulated and real brain MR images with different noise levels were segmented to demonstrate the superiority of the technique compared to other FCM-based methods.

Chang and Chung [28] designed a two-layer Hopfield neural network called the competitive Hopfield edge-finding neural network (CHEFNN) to detect the edges of CT and MRI images. To effectively remove the effect of tiny details or noises and the drawback of disconnected fractions, the CHEFNN extended the one-layer 2-D Hopfield network at the original image plane to a two-layer 3-D Hopfield network with edge detection to be implemented on its third dimension. Under the extended 3-D architecture, the network was capable of incorporating a pixel's contextual information into a pixel-labeling procedure. In addition, they [29] discovered that high-level contextual information could not be incorporated into the segmentation procedure in techniques using traditional Hopfield neural networks and thus proposed the contextual constraint-based Hopfield neural cube (CCBHNC) for image segmentation. The CCBHNC adopted a 3-D architecture with pixel classification implemented on its third dimension. Recently, still for the edge detection, Chang [30] presented a specially designed Hopfield neural network called the contextual Hopfield neural network (CHNN). The CHNN mapped the 2-D Hopfield network at the original image plane. With direct mapping, the network was capable of incorporating pixels' contextual information into an edge-detecting procedure. As a result, the CHNN could effectively remove the influence of tiny details and noise.

Most of these applications were developed based on CT or MRI images but the neural networks adopted are in quite different ways. ANN can reduce the influence of noise in the image and hence make the segmentation more robust. Further, ANN can classify different tissues and then combine them according to segmentation requirements, which is beyond the power of traditional segmentation.

1.5 Medical Image Registration with Neural Networks

Image registration is the process of transforming the different sets of data into one coordinate system. Registration is necessary in order to be able to compare or integrate the images from different measurements, which may be taken at different points in time from the same modality or obtained from the different modalities such as CT, MR, angiography, and ultrasound. Medical imaging registration often involves elastic (or nonrigid) registration to cope with elastic deformations of the body parts imaged. Nonrigid registration of medical images can also be used to register a patient's data to an anatomical atlas. Medical image registration is the preprocessing needed for many medical imaging applications with strong relevance to the result of segmentation and edge detection.

Generally, image registration algorithms can be classified into two groups: area-based methods and feature-based methods. For area-based image registration methods, the algorithm looks at the structure of the image via correlation metrics, Fourier properties, and other means of structural analysis. Most feature-based methods fine-tune their mapping to the correlation of image features: lines, curves, points, line intersections, boundaries, and so on.

To measure the volume change of lung tumor, Matsopoulos and colleagues [31] proposed an automatic, 3-D, nonrigid registration scheme that applied SOM to thoracic CT data of patients for establishing correspondence between the feature points. The practical implementation of this scheme could provide estimations of lung tumor volumes during radiotherapy treatment planning. In the algorithm, the automatic correspondence of the interpolant points

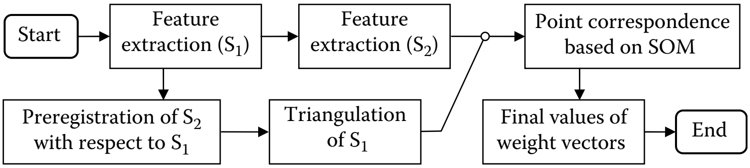


FIGURE 1.5: The elastic registration scheme.

was based on the initialization of the Kohonen neural network model able to identify 500 corresponding pairs of points approximately in the two CT sets S_1 and S_2. An overview of the described algorithm is illustrated in Figure 1.5.

In the algorithm, two sets of points were defined: S_2 is the set of points for vertebrae, ribs, and blades segmented from the reference data; and S_1 is the set of points for the same anatomical structures from the second data set, called float data. Preregistration took place between these sets of points, and triangulation of S_1 was performed. The preregistration process was applied in three dimensions and was applied in order to realign the two data sets in all coordinates. After preregistration, two steps were performed to obtain the interpolant points:

1. Triangulating S_1 and producing a wire frame based on the topology of S_1; the triangulation was based on Feitzke's work [32] and was performed by defining an SOM with the following characteristics:

    a. A grid of neurons with 20 rows by 100 columns (20×100) was chosen for the specific implementation.

    b. The initial weighting vectors of the neurons of the grid were set equal to the coordinates of a set of points extracted from an enclosing surface, typically a cylindrical surface.

    c. The input to the neural network consisted of the Cartesian coordinates of the set of points to be triangulated.

    After the process of adaptation of the neural network, the weighting vectors of the neurons had values identical to the appropriate points of S_1. A wire frame consisting of one node for each neuron could be constructed, with Cartesian coordinates of each node equal to the weight vector of the corresponding neuron. The wire frame was triangulated according to the connectivity of the neurons.

2. Establishing an SOM in terms of the topology of S_1 and training the SOM by using S_2; the search for corresponding points was based on replicating the topology of the set S_1 on the input layer of an SOM model. In the SOM model, one neuron was allocated to each node of the wire frame and the connections between the neurons were identical to the connections of the wire frame. No connection between two neurons was

accepted when the two corresponding nodes were not directly connected in the float set. The initial weight vector of the neurons was the Cartesian coordinates of the corresponding wire frame nodes in the 3-D space.

The training of the network was realized by providing the network with the coordinates of randomly selected points sampled from the reference set S_2. The neuron with weight vector closest to signal was selected to fire. The firing neuron adjusted its weight vector, and its neighboring neurons modified their weight vectors as well but less strongly. The neighboring neurons were restricted to a window of 3×3 neurons during the network training.

The convergence of the SOM network during the triangulation of S_1 set of points leads to a triangulated subset of points ($S_1{}'$). Each node of subset $S_1{}'$ corresponded to a neuron of the SOM network (20×100 neurons), whose initial weighting vector (wx_0, wy_0, wz_0) in S_1 was set to the initial Cartesian coordinates of this node. In S_1, this node was moved to new coordinates and equal to the final weighting vector (wx_1, wy_1, wz_1). The new position always coincided with a point in S_2.

Although SOM lateral interactions between neurons generated a one-to-one point correspondence, more than one point from $S_1{}'$ might correspond to one point in S_2. However, most such point mismatches are avoided by using a distance threshold criterion to exclude corresponding points exceeding a distance of more than five voxels. With the help of this process, excessive deformation of the final warped image was also prohibited. Therefore, the total number of successful corresponding points was cut down to approximately 500 pairs of points for all patient data [31].

SOM also has been used in many other applications. Shang, Lv, and Yi [33] developed an automatic method to register CT and MR brain images by using first principal directions of feature images. In the method, a PCA neural network was used to calculate the first principal directions from feature images, and then the registration was realized by aligning feature images' first principal directions and centroids.

Coppini, Diciotti, and Valli [34] presented a general approach to the problem of image matching that exploits a multiscale representation of local image structure. In the approach, a given pair of images to be matched were named target and stimulus, respectively, and were transformed by Gabor wavelets. Correspondence was calculated by exploiting the learning procedure of a neural network derived from Kohonen's SOM. The SOM neurons coincided with the pixels of the target image, and their weights were pointers to those in the stimulus images. The standard SOM rule was modified to account for image features.

Fatemizadeh, Lucas, and Soltanian-Zadeh [35] proposed a method for automatic landmark extraction from MR brain images. In the method, landmark was extracted by modifying growing neural gas (GNG), which was a neural network–based cluster-seeking algorithm. Using the modified GNG (a splitting–merging SOM), corresponding dominant points of contours extracted from two corresponding images are found. The contours were the boundaries of the regions generated by segmenting the MR brain image.

Di Bona and Salvetti [36] developed the volume-matcher 3-D project, an approach for a data-driven comparison and registration of 3-D images. The approach was based on a neural network model derived from self-organizing maps and extended to match a full 3-D data set of a source volume with the 3-D data set of a target volume.

These applications suggest that SOM is a promising algorithm for elastic registration, which is probably due to its clustering characteristics.

1.6 Other Applications with Neural Networks

In addition to those mentioned previously, ANN has been applied to other relevant areas such as medical image compression, enhancement, and restoration. In image compression [37,38], medical images such as mammograms are usually quite large in size and are stored in databases inside hospitals, which causes some difficulties in image transfer over the Internet or intranet. Some researchers applied ANN to existing compression algorithms to select interesting regions transmission for transmission or reduce the errors during the quantization in compression [40–43,47].

Panagiotidis and others [39] proposed a neural network architecture to perform lossy compression on medical images. To achieve higher compression ratio while retaining the significant (from a medical viewpoint) image content, the neural architecture adaptively selected ROI in the images.

Karlik [40] presented a combined technique for image compression based on the hierarchical finite state vector quantization and neural networks. The algorithm performed nonlinear restoration of diffraction-limited images concurrently with quantization. The neural network was trained on image pairs consisting of a lossless compression algorithm named hierarchical vector quantization.

Meyer-Bäse and colleagues [41] developed a method based on topology-preserving neural networks to implement vector quantization for medical image compression. The method could be applied to larger image blocks and represented better probability distribution estimation methods. A “neural-gas” network for vector quantization converged quickly and reached a distortion error lower than that from Kohonen’s feature map. The influence of the neural compression method on the phantom features and the mammograms was not visually perceptible up to a high compression rate.

Jaiswal and Gaikwad [42] trained a resilient backpropagation neural network to encode and decode the input data so that the resulting difference between input and output images was minimized. Lo, Li, and Freedman [43] developed a neural network–based framework to search for an optimal wavelet kernel that could be used for a specific image processing task. In the algorithm, a linear convolution neural network was applied to seek a wavelet that minimized errors and maximized compression efficiency for an image or a defined image pattern such as microcalcifications in mammograms and bone in CT head images.

To enhance original images, ANN has been used to suppress unwanted signals such as noise and tissues affecting cancerous signs. Suzuki and others [44] proposed an analysis method that makes clear the characteristics of the trained nonlinear filter, which is based on multilayer neural networks, and developed an approximate filter that achieves very similar results but was computational cost-efficient.

To detect lung nodules overlapped with ribs or clavicles in chest radiographs, Suzuki and colleagues [45] developed an image-processing technique for suppressing the contrast of ribs and clavicles in chest radiographs by means of a multiresolution massive training artificial neural network (MTANN). The structure of this neural network is illustrated in Figure 1.6, in which "bone" images are obtained by use of a dual-energy subtraction technique [46] as the teaching images to facilitate the neural network training. After that, the multiresolution MTANN was able to provide "bone-image-like" images that were similar to the teaching bone images. By subtracting the bone-image-like images from the corresponding chest radiographs, they were able to produce "soft-tissue-image-like" images where ribs and clavicles were substantially suppressed.

The MTANN consists of a linear-output multilayer ANN model, which was capable of operating on image data directly. The linear-output multilayer ANN model employed a linear function as the transfer function in the output layer because the characteristics of an ANN were improved significantly with a linear function when applied to the continuous mapping of values in image processing [47]. The inputs of the MTANN are the pixel values in a size-fixed subimage and can be written as

$$\vec{I}_{x,y} = \{I_1, I_2, \ldots, I_N\} \tag{1.5}$$

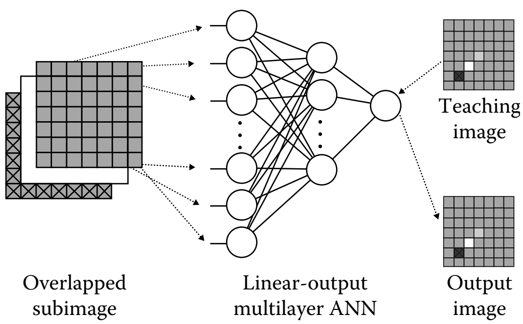


FIGURE 1.6: Architecture of MTANN.

where N is the number of inputs (i.e., the number of pixels inside a subimage). The output of the nth neuron in the hidden layer is represented by

$$O_n = f_h\left\{\sum_{m=1}^{N} w_{mn} \cdot I_m - b_n\right\} \tag{1.6}$$

where w_{mn} is a weight between the mth unit in the input layer and the nth neuron in the hidden layer, f_h is a sigmoid function, and b_n is an offset of the nth unit in the hidden layer. The output of the neuron in the output layer is represented by

$$f(x, y) = f_o\left\{\sum_{m=1}^{N_h} w_m^o \cdot O_m^H - b_o\right\} \tag{1.7}$$

where w_m^o is a weight between the mth neuron in the hidden layer and the neuron in the output layer, b_o is an offset of the neuron in the output layer, and f_o is a linear function.

To train MTANN, a dual-energy subtraction technique [48] was used to obtain the teaching image T (i.e., "bone" images) for suppression of ribs in chest radiographs. Input chest radiographs were divided pixel by pixel into a large number of overlapping subimages. Each subimage $I(x, y)$ corresponds to a pixel $T(x, y)$ in the teaching image, and the MTANN was trained with massive subimage pairs as defined in Equation 1.8:

$$\{I(x, y), T(x, y)\,|x, y \in R_T\} = \left\{\left(\vec{I}_1, T_1\right), (\vec{I}_2, T_2), \ldots, (\vec{I}_{N_T}, T_{N_T})\right\} \tag{1.8}$$

where R_T is a training region corresponding to the collection of the centers of subimages, and N_T is the number of pixels in R_T. After training, the MTANN is expected to produce images similar to the teaching images (i.e., bone-image-like images).

Since ribs in chest radiographs included various spatial–frequency components and it was difficult in practice to train the MTANN with a large subimage, multiresolution decomposition/composition techniques were employed in the algorithm. Three MTANNs for different resolutions were trained independently with the corresponding resolution images: a low-resolution MTANN was used for low-frequency components of ribs, a medium-resolution MTANN was used for medium-frequency components, and a high-resolution MTANN was used for high-frequency components. After training, the MTANNs produced a complete high-resolution image based on the images with different resolution [45].

Hainc and Kukal [49] found the ANN could also be employed as a kind of a sophisticated nonlinear filter on a local pixel neighborhood (3×3), since linear system sensitivity to impulse (isolated) noise was not good.

Chen, Chiueh, and Chen [50] introduced an ANN architecture for reducing the acoustic noise level in MRI processes. The proposed ANN consisted of two cascaded time-delay ANNs. The ANN was employed as the predictor of a feedback active noise control (ANC) system for reducing acoustic noises. Preliminary results showed that with the proposed ANC system installed, acoustic MR noises were greatly attenuated, while verbal communication during MRI sessions was not affected.

Apart from compression and enhancement, ANN has been applied to medical image processing for other purposes. Wu [51] developed a new method to extract the patient information number field automatically from the film-scanned image using a multilayer cluster neural network. Cerveri and others [52] presented a hierarchical RBF network to correct geometric distortions in X-ray image intensifiers, which reduced the accuracy of image-guided procedures and quantitative image reconstructions.

Hsu and Tseng [53] established a method to predict and create a profile of bone defect surfaces by a well-trained 3-D orthogonal neural network. To train the neural network to team the scattering characteristic, the coordinates of the skeletal positions around the boundary of bone defects were input into the network. After the neural network had been well trained, the mathematic model of the bone defect surface was generated, and the pixel positions were obtained. The 3-D orthogonal neural network avoided local minima and converges rapidly.

It is difficult to generalize all these applications of ANN into several unified models. However, it might be possible to analyze the general pattern of applying ANNs. In Section 1.7, a comparison is made by studying the applications described in all previous sections.

1.7 Conclusions

As described in the previous five sections, applications of neural networks are classified into four major categories. These applications seem quite different from one another and cover many aspects of medical image processing. To summarize all the neural networks successfully applied to medical imaging, we highlight the comparisons of their application patterns, structures, operations, and training design in Table 1.1. Because there is no theory to indicate what is the best neural network structure for medical image processing and pattern recognition, the information such as type of network, type of input, number of inputs, neurons in hidden layers, and neurons in output is listed to help with searching and designing similar neural networks for future applications. Although these applications may come from different areas, such as CAD and segmentation, and inputs for neural networks are various, the essential purpose of applying these neural networks lies in their classifications, providing inspiring summary for existing modes of neural network applications and thus leading to further developments. Since the data sets for these applications are quite different, it is not possible to compare their results and the performance

TABLE 1.1: Comparative summary of feedforward neural network applications in medical imaging.

Application	Type of Network	Purpose	Type of Input	Number of Inputs	Neurons in Hidden Layers	Neurons in Output	Train/Test/Validation
[6]	CNN*(BP*)	Detect FP*	Intensity of pixels	256	14/10	1	268ROI*/267ROI
[8]	BP	Reduce FP	Value of features	5	5	1	1448 clusters/leave-one-out
[9]	MLP*	Reduce FP	Value of features	9	20/10	1	Unknown
[10]	RBFNN*	Classify tissues	Value of features	4	5	2	44 regions/54 images
[13]	BP	Detect FP	Value of features	11	9	1	100 images/100 images/Jackknife [54]
[15]	BP	Detect FP	Value of features	10	5	1	397ROI/397 ROI/Jackknife
[17]	Feedforward	Classify boundary	Coordinate/magnitude	3	30/10	1	100 images/147 images & 65-image CV
		Classify region	Coordinate/intensity	3	50	1	
[19]	BP	Predict tissue	Value of features	8	5	1	262/leave-one-out/Jackknife
				7	3	1	
[20]	BP	Classify tissues	Value of features	3	10	1	60 primitives/983 primitives
[27]	MLP	Classify boundary	Intensity of pixels	49	30	1	1200 patterns/400 slices
[25]	BP	Classify tissues	Statistical indexes	3	Unknown	3	Small number, improved by interaction
[45]	BP	Remove noise	Intensity of pixels	25	20	1	Unknown
[46]	MTANN*(BP)	Classify tissues	Intensity of pixels	81	20	1	5000 regions/118 images
[55]	BP	Classify MC	Value of features	14	13	1	100 ROI/leave-one-out
[56]	MLBNN*(BP)	Classify MC	Vectors from SOM	5	25/14	7	32 cases/64 cases
[57]	BP	Classify tissues	Vectors from SOM	3	7	7	Unknown/80 images
[58]	BP	Detect edge	Intensity of pixels	121	20	1	24 images/fourfold CV

BP, backpropagation; CNN, convolution neural network; CV, cross-validation; FP, false positive MC or regions; MC, microcalcification cluster; MLBNN, multilayered BP neural network; MLP, multilayer perceptron; RBFNN, radial basis function neural network; ROI, region of interest; SOM, self-organizing map; MTANN, massive training artificial neural network.

of these algorithms. Some applications are ignored in the list because the details about their neural networks are limited. The total number of neurons needed in the hidden layers somewhat depends on the total number of training samples.

In contrast to feedforward neural network, the applications of feedback neural networks for medical image processing have been quite limited in the past decade, and most of them are in the area of image segmentation and are primarily based on Hopfield neural networks. The similarities among these applications are quite limited, but all of them need to minimize an energy function during convergence of the network. The energy function must be designed individually, which might affect its application in medical imaging. Because the Hopfield neural network is unsupervised, it may not work for CAD like the feedforward neural network, which requires a priori knowledge in classifications.

Although the applications of Kohonen's SOM are not as many as those of feedforward neural networks, its clustering and unsupervised properties make it very suitable for image registration. SOM converges to a solution that approximates its input data by adapting to prototype vectors. During this process, the relation of its neighborhood neurons is also taken into account, leading to preservation of topology and mapping of training sets. For the applications of image registration, the input vectors of the neurons in SOM usually contain the spatial coordinate and intensity of pixels. For applications in image compression, SOM is used as a topology-preserving feature map to generate vector quantization for code words. Sometimes, SOM produces the segmentation results for feedforward neural networks due to its unsupervised clustering property.

In summary, the applications of ANN in medical image processing have to be analyzed individually, although many successful models have been reported in the literature. ANN has been applied to medical images to deal with the issues that cannot be addressed by traditional image processing algorithms or by other classification techniques. By introducing ANNs, algorithms developed for medical image processing and analysis often become more intelligent than conventional techniques. While this chapter provided a focused survey on a range of neural networks and their applications to medical imaging, the main purpose here is to inspire further research and development of new applications and new concepts in exploiting neural networks.

Acknowledgment

The authors wish to acknowledge the financial support from the British Cancer Research Trust, through which this survey was carried out as part of a funded research project.

References

[1] Ripley, B. D. 1996. Feed-forward neural networks. In *Pattern Recognition and Neural Networks*, 143–80. Cambridge: Cambridge University Press.

[2] Wasserman, P. D. 1993. Backpropagation and beyond. In *Advanced Methods in Neural Computing*, 97–118. New York: Van Nostrand Reinhold.

[3] Wasserman, P. D. 1993. Radial basis function networks. In *Advanced Methods in Neural Computing*, 147–176. New York: Van Nostrand Reinhold.

[4] Carling, A. 1992. Recurrent networks. In *Introducing Neural Networks*, 189–94. Wilmslow: Sigma.

[5] Kohonen, T. 1990. The self-organizing map. *Proceedings of the IEEE* 78: 1464–80.

[6] Ge, J., B. Sahiner, L. M. Hadjiiski, H. P. Chan, J. Wei, M. A. Helvie, and C. Zhou. 2006. Computer-aided detection of clusters of microcalcifications on full field digital mammograms. *Medical Physics* 33: 2975–88.

[7] Lo, S. C. B., H. P. Chan, J. S. Lin, H. Li, M. T. Freedman, and S. K. Mun. 1995. Artificial convolution neural network for medical image pattern recognition. *Neural Networks* 8: 1201–14.

[8] Nagel, R. H., R. M. Nishikawa, J. Papaioannou, and K. Doi. 1998. Analysis of methods for reducing false positives in the automated detection of clustered microcalcifications in mammograms. *Medical Physics* 25: 1502–6.

[9] Papadopoulossa, A., D. I. Fotiadisb, and A. Likasb. 2002. An automatic microcalcification detection system based on a hybrid neural network classifier. *Artificial Intelligence in Medicine* 25: 149–67.

[10] Christoyiani, I., E. Dermatas, and G. Kokkinakis. 2000. Fast detection of masses in computer-aided mammography. *IEEE Signal Processing Magazine* 17: 54–64.

[11] Patrocinio, A. C., H. Schiabel, R. H. Benatti, C. E. Goes, and F. L. S. Nunes. 2000. Investigation of clustered microcalcification features for an automated classifier as part of a mammography CAD scheme. *Proceedings of the 22nd Annual EMBS International Conference* 1203–5.

[12] Setiono, R. 1996. Extracting rules from pruned neural networks for breast cancer diagnosis. *Artificial Intelligence in Medicine* 8: 37–51.

[13] Xu, X. W., K. Doi, T. Kobayashi, H. MacMahon, and M. L. Giger. 1997. Development of an improved CAD scheme for automated detection of lung nodules in digital chest images. *Medical Physics* 24: 1395–1403.

[14] Zhou, Z. H., Y. Jiang, Y. B. Yang, and S. F. Chen. 2002. Lung cancer cell identification based on artificial neural network ensembles. *Artificial Intelligence in Medicine* 24: 25–36.

[15] Keserci, B., and H. Yoshida. 2002. Computerized detection of pulmonary nodules in chest radiographs based on morphological features and wavelet snake model. *Medical Image Analysis* 6: 431–47.

[16] Qian, W., T. Zhukov, D. S. Song, and M. S. Tockman. 2007. Computerized analysis of cellular features and biomarkers for cytologic diagnosis of early lung cancer. *Analytical and Quantitative Cytology and Histology* 29: 103–11.

[17] Coppini, G., S. Diciotti, M. Falchini, N. Villari, and G. Valli. 2003. Neural networks for computer-aided diagnosis: Detection of lung nodules in chest radiograms. *IEEE Transactions on Information Technology in Biomedicine* 7: 344–57.

[18] Mohamed, E. I., C. Maiolo, R. Linder, S. J. Poppl, and A. De Lorenzo. 2003. Artificial neural network analysis: A novel application for predicting site-specific bone mineral density. *Acta Diabetologica* 40: S19–22.

[19] Scott, J. A. 2004. The lung scan and the abnormal chest X-ray: difficult diagnoses. *Nuclear Medicine Communications* 25: 1137–41.

[20] Kobashi, S., N. Kamiura, Y. Hata, and F. Miyawaki. 2001. Volume-quantization-based neural network approach to 3D MR angiography image segmentation. *Image and Vision Computing* 19: 185–93.

[21] Middleton, I., and R. I. Damper. 2004. Segmentation of magnetic resonance images using a combination of neural networks and active contour models. *Medical Engineering & Physics* 26: 71–86.

[22] Lin, J. S. 2000. Segmentation of medical images through a penalized fuzzy Hopfield network with moments preservation. *Journal of the Chinese Institute of Engineers* 23: 633–43.

[23] Lin, K. C. R., M. S. Yang, H. C. Liu, J. F. Lirng, and P. N. Wang. 2003. Generalized Kohonen's competitive learning algorithms for ophthalmological MR image segmentation. *Magnetic Resonance Imaging* 21: 863–70.

[24] Dokur, Z., and T. Olmez, 2003. Segmentation of MR and CT images by using a quantiser neural network. *Neural Computing & Applications* 11: 168–77.

[25] Stalidis, G., N. Maglaveras, S. N. Efstratiadis, A. S. Dimitriadis, and C. Pappas. 2002. Model-based processing scheme for quantitative 4-D cardiac MRI analysis. *IEEE Transactions on Information Technology in Biomedicine* 6: 59–72.

[26] Chang, C. L., and Y. T. Ching. 2002. Fuzzy Hopfield neural network with fixed weight for medical image segmentation. *Optical Engineering* 41: 351–58.

[27] Shen, S., W. Sandham, M. Granat, and A. Sterr. 2005. MRI fuzzy segmentation of brain tissue using neighborhood attraction with neural-network optimization. *IEEE Transactions on Information Technology in Biomedicine* 9: 459–67.

[28] Chang, C. Y., and P. C. Chung. 2000. Two-layer competitive based Hopfield neural network for medical image edge detection. *Optical Engineering* 39: 695–703.

[29] Chang, C. Y., and P. C. Chung. 2001. Medical image segmentation using a contextual-constraint-based Hopfield neural cube. *Image and Vision Computing* 19: 669–78.

[30] Chang, C. Y. 2006. Contextual-based Hopfield neural network for medical image edge detection. *Optical Engineering* 45: Art. No. 037006.

[31] Matsopoulosa, G. K., N. A. Mouravlianskya, P. A. Asvestasa, K. K. Delibasisa, and V. Koulouliasb. 2005. Thoracic non-rigid registration combining self-organizing maps and radial basis functions. *Medical Image Analysis* 9: 237–54.

[32] Fritzke, B. 1994. A growing neural gas network learns topologies. In G. Tesauro, D. S. Touretzky, and T. L. Leen, eds., *Advances in Neural Information Processing Systems*, 7th ed., 625–32. Cambridge, MA: MIT Press.

[33] Shang, L. F., J. C. Lv, and Z. Yi. 2006. Rigid medical image registration using PCA neural network. *Neurocomputing* 69: 1717–22.

[34] Coppini, G., S. Diciotti, and G. Valli. 2004. Matching of medical images by self-organizing neural networks. *Pattern Recognition Letters* 25: 341–52.

[35] Fatemizadeh, E., C. Lucas, and H. Soltanian-Zadeh. 2003. Automatic landmark extraction from image data using modified growing neural gas network. *IEEE Transactions on Information Technology in Biomedicine* 7: 77–85.

[36] Di Bona, S., and O. Salvetti. 2002. Neural method for three-dimensional image matching. *Journal of Electronic Imaging* 11: 497–506.

[37] Bovik, A. C. 2005. *Handbook of Image and Video Processing*. London: Academic Press.

[38] Jiang, J. 1999. Image compression with neural networks: A survey. *Signal Processing—Image Communication* 14: 737–60.

[39] Panagiotidis, N. G., D. Kalogeras, S. D. Kollias, and A. Stafylopatis. 1996. Neural network-assisted effective lossy compression of medical images. *Proceedings of the IEEE* 84: 1474–87.

[40] Karlik, B. 2006. Medical image compression by using vector quantization neural network (VQNN). *Neural Network World* 16: 341–48.

[41] Meyer-Bäse, A., K. Jancke, A. Wismüller, S. Foo, and T. Martinetz. 2005. Medical image compression using topology-preserving neural networks. *Engineering Applications of Artificial Intelligence* 18: 383–92.

[42] Jaiswal, R. R., and A. N. Gaikwad. 2006. Neural network assisted effective lossy compression of medical images. *IETE Technical Review* 23: 119–26.

[43] Lo, S. C. B., H. Li, and M. T. Freedman. 2003. Optimization of wavelet decomposition for image compression and feature preservation. *IEEE Transactions on Medical Imaging* 22: 1141–51.

[44] Suzuki, K., I. Horiba, and N. Sugie. 2002. Efficient approximation of neural filters for removing quantum noise from images. *IEEE Transactions on Signal Processing* 50: 1787–99.

[45] Suzuki, K., H. Abe, H. MacMahon, and K. Doi. 2006. Image-processing technique for suppressing ribs in chest radiographs by means of massive training artificial neural network (MTANN). *IEEE Transactions on Medical Imaging* 25: 406–16.

[46] Suzuki, K., I. Horiba, K. Ikegaya, and M. Nanki. 1995. Recognition of coronary arterial stenosis using neural network on DSA system. *Systems and Computers in Japan* 26: 66–74.

[47] Suzuki, K. 2003. Neural edge enhancer for supervised edge enhancement from noisy images. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 25: 1582–96.

[48] Ishigaki, T., S. Sakuma, Y. Horikawa, M. Ikeda, and H. Yamaguchi. 1986. One-shot dual energy subtraction imaging. *Radiology* 161: 271–73.

[49] Hainc, L., and J. Kukal. 2006. Role of robust processing in ANN de-noising of 2D image. *Neural Network World* 16: 163–76.

[50] Chen, G. K., T. D. Chiueh, and J. H. Chen. 1999. Active cancellation system of acoustic noise in MR imaging. *IEEE Transactions on Biomedical Engineering* 46: 186–91.

[51] Wu, H. H. P. 2004. Patient information extraction in digitized X-ray imagery. *Image and Vision Computing* 22: 215–26.

[52] Cerveri, P., C. Forlani, N. A. Borghese, and G. Ferrigno. 2002. Distortion correction for x-ray image intensifiers: Local unwarping polynomials and RBF neural networks. *Medical Physics* 29: 1759–71.

[53] Hsu, J. H., and C. S. Tseng. 2001. Application of three-dimensional orthogonal neural network to craniomaxillary reconstruction. *Computerized Medical Imaging and Graphics* 25: 477–82.

[54] Dorfman, D. D., K. S. Berbaum, and C. E. Metz. 1992. Receiver operating characteristic rating analysis: Generalization to the population of readers and patients with the jackknife method. *Investigative Radiology* 27: 723–31.

[55] Kallergi, M. 2004. Computer-aided diagnosis of mammographic microcalcification clusters. *Medical Physics* 31: 314–26.

[56] Glass, J. O., W. E. Reddick, O. Goloubeva, V. Yo, and R. G. Steen. 2000. Hybrid artificial neural network segmentation of precise and accurate inversion recovery (PAIR) images from normal human brain. *Magnetic Resonance Imaging* 18: 1245–53.

[57] Reddick, W. E., R. K. Mulhern, T. D. Elkin, J. O. Glass, T. E. Merchant, and J. W. Langston. 1998. A hybrid neural network analysis of subtle brain volume differences in children surviving brain tumors. *Magnetic Resonance Imaging* 16: 413–21.

[58] Suzuki, K., I. Horiba, N. Sugie, and M. Nanki. 2004. Extraction of left ventricular contours from left ventriculograms by means of a neural edge detector. *IEEE Transactions on Medical Imaging* 23: 330–39.

Chapter 2

Evolutionary Computing and Its Use in Medical Imaging

Lars Nolle and Gerald Schaefer

Contents

2.1 Evolutionary Computing 28
2.1.1 Systems 28
2.1.2 Objective function 29
2.1.3 Search space and fitness landscape 29
2.1.4 Optimization 30
2.1.5 Optimization loop 30
2.2 Genetic Algorithms 31
2.2.1 Selection 34
2.2.2 Crossover 35
2.2.3 Mutation 36
2.2.4 Discussion 36
2.2.5 Schemata theorem 37
2.2.6 Coding problem 38
2.3 Genetic Programming 39
2.3.1 Selection 40
2.3.2 Crossover 40
2.3.3 Mutation 41
2.4 Evolutionary Approaches for Medical Imaging 42
2.5 Conclusions 43
References 43

Many scientific problems can be formulated as optimization problems. Among the many classes of algorithms for solving such problems, one interesting, biologically inspired group is that of evolutionary optimization techniques. In this chapter, we provide an overview of such techniques, in particular of genetic algorithms and genetic programming and its related subtasks of selection, crossover, mutation, and coding. We then explore some applications of genetic techniques in the context of medical imaging.

2.1 Evolutionary Computing

Many scientific problems can be viewed as search or optimization problems, where an optimum input parameter vector for a given system has to be found in order to maximize or to minimize the system response to that input vector. Often, auxiliary information about the system, like its transfer function and derivatives, is not known, and the measures might be incomplete and distorted by noise. This makes such problems difficult to solve by traditional mathematical methods. Evolutionary optimization algorithms, which are based on biological principles borrowed from nature, can offer a solution. These algorithms work on a population of candidate solutions, which are iteratively improved so that an optimal solution evolves over time.

This chapter discusses the general problem of search and optimization before it introduces the system's view, followed by a definition of search space and fitness landscape. It then explains the process of optimization and the concept of optimization loops. It introduces biologically inspired evolutionary optimization algorithms: genetic algorithms and genetic programming. Finally, it provides an overview of same sample applications of evolutionary approaches in medical imaging.

2.1.1 Systems

Every process or object can be seen as a system. Fenton and Hill (1993) define a system as "an assembly of components, connected together in an organised way, and separated from its environment by a boundary. This organised assembly has an observable purpose which is characterised in terms of how it transforms input from the environment into output to the environment." By definition, a system has exactly one input channel x and exactly one output channel y (see Figure 2.1). All interactions with the environment must be made through these interfaces.

Both input and output can be vectors or scalars. The input is called the *independent variable* or *parameter*, because its value(s) can be chosen freely, and it results in the output y, the so-called *dependent variable*. If the present state of the system does not depend on previous states but only on the current input, the system is said to be a *steady-state system*, and the output of the system can be described as a function of the input $y = f(x)$.

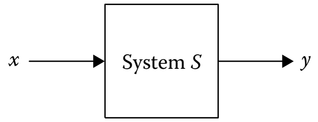


FIGURE 2.1: Generic system.

2.1.2 Objective function

In order to rate the quality of a candidate solution x, it is necessary to transform the system response to x into an appropriate measure, called the *objective* or *fitness*. If the system has only one output variable, the system output y equals the fitness. If y has more than one component, the output variables of the system have to be combined into a single value, computed by the *objective function* or *fitness function*. In general, there are four approaches to judge the system output: aggregation, the changing objectives method, the use of niche techniques, and Pareto-based methods (Fonseca and Fleming, 1995). The most often used method is aggregation. In its simplest case, the fitness function $F(x)$ equals the weighted sum of the components $y_i = c_i \cdot F_i(x)$ of y, where c_i is the weight for component i:

$$F(x) = c_0 + c_1 \cdot F_1(x) + \cdots + c_n \cdot F_n(x) \tag{2.1}$$

2.1.3 Search space and fitness landscape

If all the possible candidate solutions are collected in an ordered way, this collection is called the search space and sometimes the input space. For an optimization problem of dimension n, that is, a system with n independent parameters, the search space also has dimension n. Adding the dimension *fitness* or *costs* to the search space results in the $(n + 1)$ dimensional fitness landscape (Wright, 1931); see Figure 2.2.

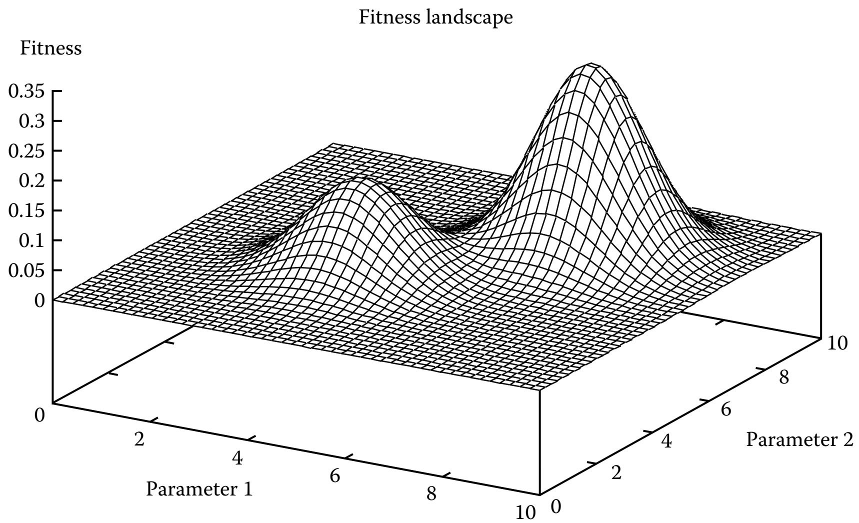


FIGURE 2.2: Example of a fitness landscape for a system with two input parameters.

2.1.4 Optimization

Optimization (Schwefel, 1995) is the process of selecting the best candidate solution from a range of possibilities (i.e., from the search space). In other words, a system S that has to be optimized in terms of a quality output value y is brought into a new state that has a better quality output value y than the previous state. This is done by changing the independent input parameters x. The error function describes the difference between the predefined objective $y_{desired}$ and system response $f(x)$ to the input x.

$$Error(x) = y_{desired} - f(x) \tag{2.2}$$

Usually, the aim is to find the vector x' that leads to a minimal error for the system S, that is, the minimal departure from the optimal output value:

$$Error(x') = 0 \tag{2.3}$$

Often, a predefined target value is not known. In this case, one tries to gain a fitness value that is as high as possible in the case of maximization or as low as possible in the case of minimization. Ideally, one would evaluate all possible candidates and choose the best one. This is known as exhaustive search. However, often it is not feasible to consider all possible solutions, for example, if the search space is too large and the evaluation of a single candidate is too expensive. In such cases, only a subset of the solutions can be evaluated.

Optimization problems can be either function optimization problems or combinatorial problems. The first class of problems can be divided into continuous optimization and discrete optimization problems. In continuous function optimization, the independent variables are real numbers, whereas for discrete function optimization, the independent variables can be chosen only from a predefined set of allowed and somehow ordered numbers, such as $\{10, 20, 30, 40\}$.

In combinatorial optimization problems, the optimum sequence or combination of a fixed set of input values has to be found. Here, the input values are symbols and might not be connected or ordered, for example {apple, orange, strawberry}. An example of a combinatorial optimization problem is the classical traveling salesman problem (TSP), where a sales agent needs to visit a predefined set of cities and return to base. The problem here is to find an optimal route that connects all cities while having the shortest travel distance by choosing the order in which the cities are visited.

2.1.5 Optimization loop

Mathematical or calculus-based methods use known functional relationships among variables and objectives to calculate the optimum of the given system. Therefore, an exact mathematical model of the process must exist. Edelbaum (1962) introduced the differentiation of calculus-based methods in direct methods and indirect methods.

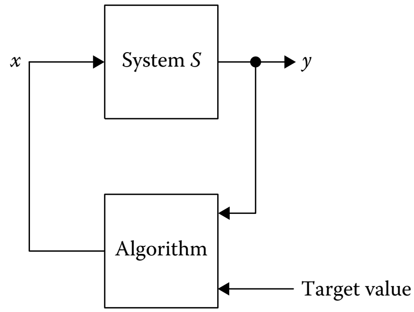


FIGURE 2.3: Closed optimization loop consisting of a system and an optimization algorithm.

Direct methods solve the optimization problem by iterative calculation and derivation of the error function and moving in a direction to the maximum slope gradient. Indirect methods solve the optimization problem in one step—without testing—by solving a set of equations (usually nonlinear). These equations result from setting the derivative of the error function equal to zero.

Both classes of methods are local in scope: they tend to find only local optima. Therefore, they are not robust. They depend on the existence of derivatives. Real problem functions tend to be perturbed by noise and are not smooth (i.e., derivations may not exist for all points of functions). This class of problem cannot be solved by mathematical methods.

If the functional relations among input variables and objectives are not known, one can experiment on the real system (or a model of this system) in order to find the optimum. Access to the independent variables must exist for the whole multidimensional search space—the collection of all possible candidate solutions. Also, a possibility of measuring the independent variable and the objective must be given. The optimization process is iterative; that is, it has to be done in a closed optimization loop (Figure 2.3).

Experimental optimization methods can therefore be seen as a search for the optimum by traversing over the fitness landscape.

2.2 Genetic Algorithms

As Darwin's theory of natural selection articulates, nature is very effective at optimization (e.g., enabling life forms to survive in a unfriendly and changing environment by means of simple trial and error). Genetic algorithms (GAs) simulate this evolutionary mechanism by using heredity and mutation. They were first introduced by Holland (1975), who also provided a theoretical framework for genetic algorithms, the schemata theorem (Goldberg, 1989).

For genetic algorithms, the independent input parameters of a system S (Figure 2.4) are coded into a binary string, the *genotype* of an individual (Figure 2.5).

The individual represented by genotype is called a *phenotype*. This phenotype has a certain quality or fitness to survive, which can be determined by presenting the phenotype to the system S and measuring the system response.

The search is undertaken not only by one individual but by a population of n genotypes, the *genepool* (Figure 2.6). Therefore, the search space is tested at n points in parallel. All the individuals of the genepool at a time t_n are called a *generation*.

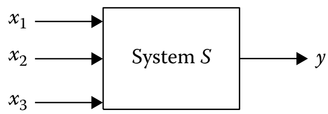


FIGURE 2.4: System to be optimized.

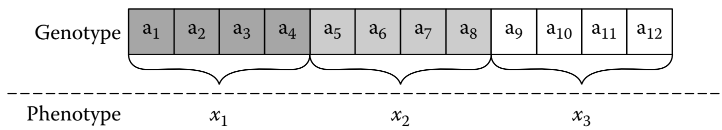


FIGURE 2.5: Binary string representing one input pattern of the system.

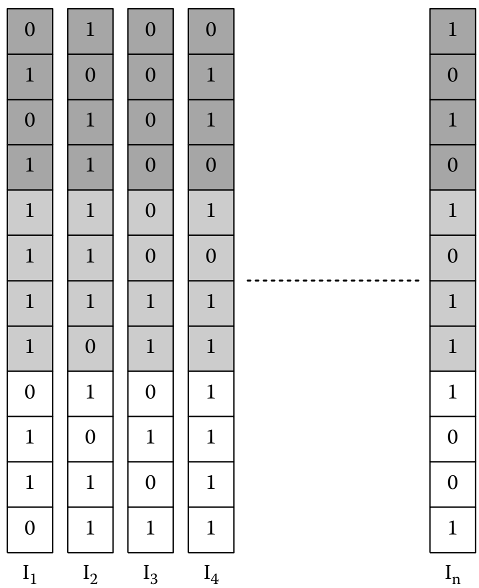


FIGURE 2.6: Genepool consisting of individuals $I_1 \ldots I_n$.

A new generation for time t_{n+1} is generated by selecting N individuals from the current population for breeding. They are copied into the genepool of the next generation, and their genetic information is then recombined, using the *crossover* operator (see Section 2.2.2), with a predefined crossover probability p_c. The resulting offspring is then copied into the new genepool, and mutation is applied to the offspring. Figure 2.7 shows the flowchart of a simple genetic algorithm.

The search is carried out until at least one individual has a better fitness than the defined minimum fitness or a maximum number of generations has been reached.

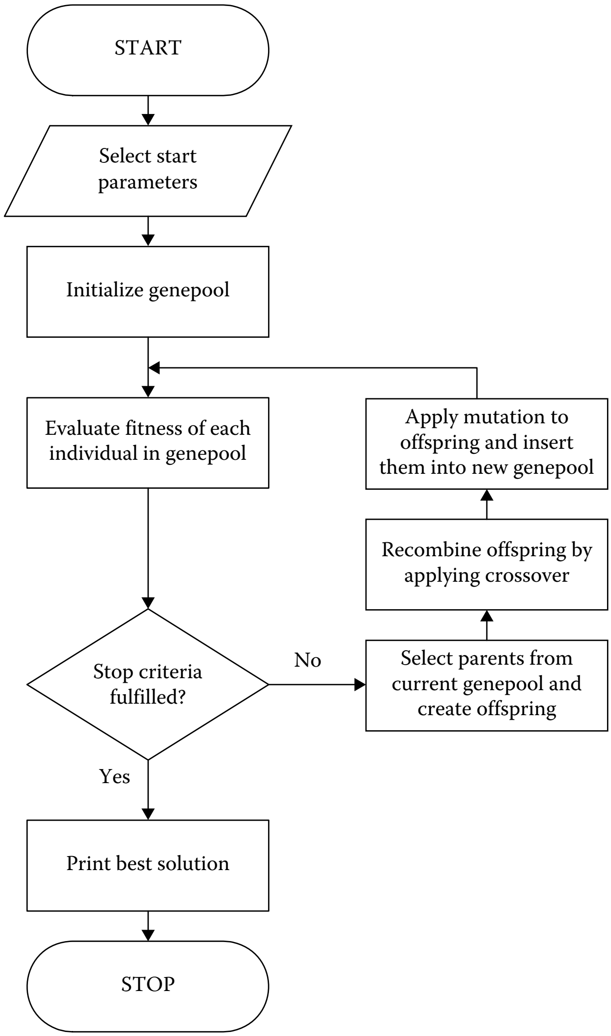


FIGURE 2.7: Flowchart of basic GA algorithm.

2.2.1 Selection

In general, there are three approaches to choose individuals from the current generation for reproduction: tournament selection, fitness proportional selection, and rank-based selection. In tournament selection, two or more individuals are randomly selected from the current generation of N genotypes to compete with each other. The individual with the highest fitness of this set is the winner and is selected for generating offspring. The process is repeated N times in order to create the new population. Using tournament selection, the least fit individual can never be selected.

In fitness proportional selection, the chance of an individual to be selected is related to its fitness value. The most commonly used method of this type is roulette wheel selection in which proportions of an imaginary roulette wheel are distributed in proportion to the relative fitness of an individual. Figure 2.8 shows an example for $N = 3$. In this example, the fitness of individual 3 is approximately four times higher than the fitness of individual 1, which means its chance of selection is four times greater than that of individual 1. For a population of N individuals, the wheel is spun N times, and the individual under the pointer is selected. In fitness proportional selection, all individuals have a chance of selection, but high-fitness individuals are more likely to be selected because they occupy a larger portion of the wheel.

However, there is the statistical chance that the actual selected distribution might not reflect the expected distribution based on the fitness values. If the selection is too strong, it can lead to premature convergence: the population would converge before it found the region of the search space that contains the global optimum. In other words, the exploitation would start before the search space is fully explored. On the other hand, if the selection is too weak, it can lead to stalled evolution, which means the search is reduced to randomly walking through search space.

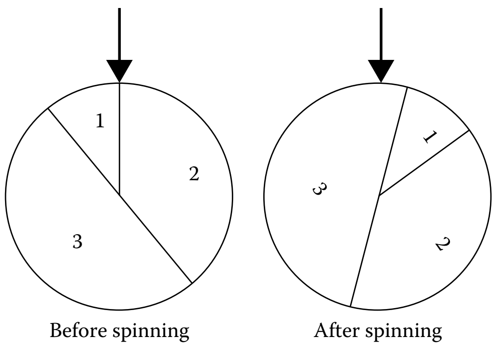


FIGURE 2.8: Roulette wheel selection.

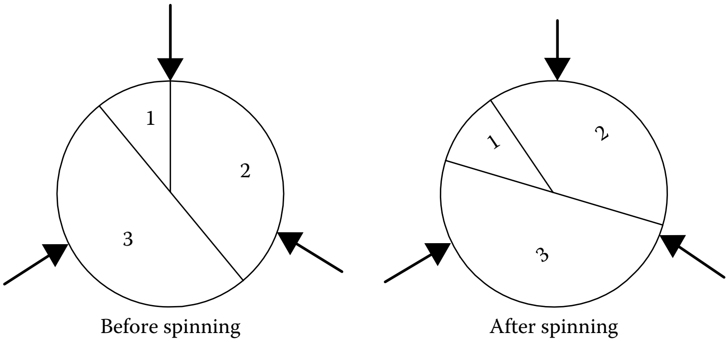


FIGURE 2.9: SUS selection.

These effects are overcome using stochastic universal selection (SUS). Here, the same roulette wheel is used, but instead of using a single pointer, N equally spaced pointers are used for a population of N individuals, and the wheel is spun only once (Figure 2.9).

Instead of using the fitness of an individual for selection, a selective s value can be used, which is based on the rank position of an individual in the population (Equation 2.4).

$$s_i = Min + (Max - Min)\frac{rank_i - 1}{N - 1} \tag{2.4}$$

where

Min: minimum fitness within a generation
Max: maximum fitness within a generation
$rank_i$: rank of individual i within the population in a generation
N: number of individuals within population

So, instead of using the raw fitness to determine the proportion for an individual, the rank of the individual within the generation is used.

Sometimes the m fittest individuals in a generation are cloned into the next generation in order to preserve their genetic material. This is known as elitism.

2.2.2 Crossover

The most important operator in terms of robustness of the algorithm is the crossover operator. Figure 2.10 shows the one-point crossover operator, which combines the information of two parents. They are aligned and then both cut at a randomly chosen crossover point, and the tails are swapped successively.

Instead of a single crossover point, two or more random crossover points can be used for recombining the genetic information of the parents.

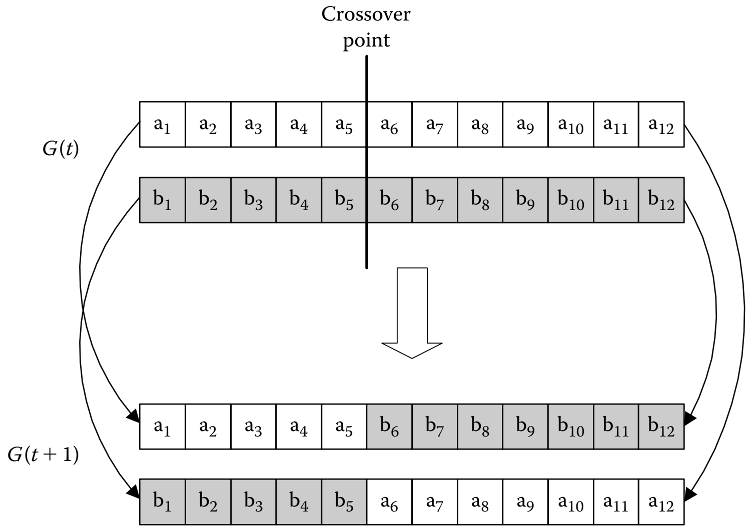


FIGURE 2.10: Crossover operator.

Another form of crossover is called uniform crossover (Syswerda, 1989). Here, every component of a parent individual X is randomly passed on either to offspring A or offspring B. If X passes on its component to A, the position in B is filled using the component from parent Y, and vice versa.

2.2.3 Mutation

After the genetic information of the parents is recombined using crossover, mutation is applied to every individual of the new generation. Every bit of the offspring is inverted (mutated) with probability p_m. The mutation operator is important for restoring lost information and producing a better effectiveness of the genetic algorithm.

2.2.4 Discussion

The advantages of genetic algorithms are that they use payoff (objective function) information, not derivatives or other auxiliary knowledge; that is, they are black-box optimization methods. Genetic algorithms tend to converge toward the global optimum rather than getting stuck in a local optimum, and therefore they are very robust. On the other hand, it is not always straightforward to find the right GA parameters for a particular optimization problem, such as a suitable genepool size or mutation probability. Also, the efficiency of genetic algorithms relies heavily on the right coding of the input parameters (i.e., the chosen mapping function from phenotype to genotype), and they tend to fail if the inputs of the system are heavily correlated.

2.2.5 Schemata theorem

Holland provided a theoretical foundation of genetic algorithms—a theoretical proof of convergence—which he called the schemata theorem. A schema is a template for binary strings, but built from a three-letter alphabet containing the symbols *, 0, and 1. The * symbol is the "don't care" symbol, which stands for either 0 or 1. Figure 2.11 shows an example of a schema for chromosomes consisting of 12 bits, of which 3 are set to the don't care symbol and the remaining 9 bits are set to fixed values.

The distance between the first and the last fixed bit is called the defined length of the schema, and the number of fixed bits is called the order of the schema. Figure 2.12 shows an example of a schema H and the different instances it represents.

A binary string s is an instance of a schema H if it fits into the template. Therefore, any binary string of length l does not represent just one candidate solution; it is simultaneously an instance of 2^1 schemata. Consequently, a genetic algorithm with the genepool of size n tests not only n different solutions but also a high number of different schemata at the same time. This is known as implicit parallelism in genetic algorithm and provides an explanation for their effectiveness and efficiency.

According to Holland, the number of instances m of a schema H that are contained in the population at generation $t+1$ can be determined as follows:

$$m(H, t+1) = m(H, t) \cdot \frac{\bar{f}(H)}{\bar{f}} \tag{2.5}$$

1	1	*	1	0	0	0	*	*	1	1	0

FIGURE 2.11: Example of a schema in GA.

1	*	*	0	Schema *H*
1	0	0	0	Instance 1
1	0	1	0	Instance 2
1	1	0	0	Instance 3
1	1	1	0	Instance 4

FIGURE 2.12: Example of a schema H and the instance it represents.