TAMPEREEN TEKNILLINEN YLIOPISTO
TAMPERE UNIVERSITY OF TECHNOLOGY
Julkaisu 652 • Publication 652

Alireza Akhbardeh

Signal Classification Using Novel Pattern Recognition Methods and Wavelet Transforms

Tampere 2007

Tampereen teknillinen yliopisto. Julkaisu 652
Tampere University of Technology. Publication 652

Alireza Akhbardeh

Signal Classification Using Novel Pattern Recognition Methods and Wavelet Transforms

Thesis for the degree of Doctor of Technology to be presented with due permission for public examination and criticism in Tietotalo Building, Auditorium TB109, at Tampere University of Technology, on the 9th of March 2007, at 12 noon.

Tampereen teknillinen yliopisto - Tampere University of Technology
Tampere 2007

ISBN 978-952-15-1721-1 (printed)
ISBN 978-952-15-1740-2 (PDF)
ISSN 1459-2045

Signal Classification Using Novel Pattern Recognition Methods and Wavelet Transforms

Alireza Akhbardeh
Institute of Signal Processing
Tampere University of Technology
alireza.akhbardeh@tut.fi

Supervisors:

- **Professor Moncef Gabbouj**, Institute of Signal Processing, Tampere University of Technology, Finland
- **Docent Alpo Värri**, Institute of Signal Processing, Tampere University of Technology, Finland

Pre-examiners:

- **Professor Michalis Zervakis**, Digital Image and Signal Processing Laboratory, Technical University of Crete, Greece
- **Dr. Ricardo Vigario**, Laboratory of Computer and Information Science, Helsinki University of Technology, Finland

Opponent:

Professor Rangaraj Rangayyan, Department of Electrical and Computer Engineering, University of Calgary, Canada

To Arezou with all my love and to my parents

Abstract

A complete pattern recognition system consists of a sensor that gathers the observations to be classified or described; a feature extraction mechanism that computes numeric or symbolic information from the observations; and a classification or description scheme that does the actual job of classifying or describing observations, relying on the extracted features. A pattern recognition example, in this dissertation, is the Ballistocardiogram (BCG). The BCG measurement, recording systems, and signal pre-processing were studied as part of the work. The thesis reviews various BCG measurement techniques and devices, noise removal from the measurements and segmentation methods of the BCG signal.

Different types of wavelet transforms (WTs), as feature extraction methods, were studied and applied for the classification of BCG. A novel feature extraction method called 'Time-frequency moments singular value decomposition (TFM-SVD)' was also developed yielding results similar to the WT.

The development of machine learning algorithms is essential in developing intelligent systems such as autonomous robots. Artificial neural networks (ANNs) are one of the technologies in learning systems. Usually the learning process is based on training ANNs with a representative set of real world examples and then the trained network is embedded into a system. There are, however, a number of problems with most existing ANN structures. These include time consuming training, large amounts of training data and the fact that complicated structures are difficult to implement in embedded systems and integrated circuits, in particular.

The aim of the study was to address the above problems by developing novel methods for well-known pattern classification test data sets such as IRIS and Vowel data as well as for BCG. The developed learning algorithms (QuickLearn, CombilNet and its example SF-ART) performed equally well in pattern classification performance with conventional ANNs although SF-ART required less than ten training cycles. The QuickLearn algorithm classifies data almost as well as the traditional ANNs although it requires only one learning cycle.

Preface

The research reported in this thesis has been carried out in the Institute of Signal Processing at Tampere University of Technology during October 2004-Febraury/March 2007. However, the foundations of this work were from the Proactive Health Monitoring Project, sponsored by the Academy of Finland 2003-2005 where I worked as a research scientist during my studies at doctoral level. My supervisors were Professor Moncef Gabbouj and Docent Alpo Värri (PhD) who introduced me to the world of pattern recognition and signal processing. For this and for their invaluable advice and guidance, I am indebted to them. I am also deeply grateful to professors specially Professors Jaakko Astola and Moncef Gabbouj for welcoming me to the Institute of Signal Processing as well as for their support and encouragement.

The reviewers of this thesis, Professor Michalis Zervakis (Digital Image and Signal Processing Laboratory, Technical University of Crete, Greece) and Dr. Ricardo Vigario (Laboratory of Computer and Information Science, Helsinki University of Technology, Finland), deserve my heartfelt thanks for their careful reading and constructive feedback of the manuscript of this thesis.

I wish to thank my colleagues, Sakari Junnila, Mikko Koivuluoma and Laurentiu Barna from our Institute and Teemu Koivistoinen, Tiit Kööbib, from Tampere University Hospital and all the other past and present members of the Sleep and Sensory Signal Analysis research group who have provided their assistance when I needed it. I sincerely thank all my colleagues in the Institute Signal of Processing for providing a stimulating working environment.

I would like to express my boundless thanks to Arezou for her love and understanding during all these years and never ending support particularly during the final stage of this thesis. Without her sacrifices and encouragement, this work would never have been finished. I truly dedicate this work to her. My warmest thanks to my beloved parents specially my mother, my father, my brothers, and my sister, for their encouragement and love throughout my life and studies.

Finally, I want to express my gratitude towards the organizations that have financially supported this work. They are Academy of Finland, Tampere Graduate School of Information, Science and Engineering (TISE) and Centre for International Mobility (CIMO).

Tampere, September 21, 2006 Alireza Akhbardeh

Table of Contents

List of Figures

List of Tables

List of Abbreviations

ANN	Artificial Neural Networks
ART	Adaptive Resonance Theory
AUC	Area under ROC
BCG	Ballistocardiogram
Biorth	Biorthogonal wavelets
CI	Confidence Interval
CNeT	Competitive Neural Trees
CombilNet	Combined Learning Network
CombiMap (CMaP)	Combined Map
DB	Daubecheis Wavelets
ECG	Electrocardiogram
MLP	Multi-Layer Perceptrons
Nh	Number of hidden layer
ProHeMon	Proactive Health Monitoring
QL	QuickLearn Algorithm
RBF	Radial Basis Functions
ROC	Receiver Operating Characteristic
SF-ART	Supervised Fuzzy Adaptive Resonance Theory
SV	Singular Value
SVM	Support Vector Machine
TFM-SVD	Time Frequency Moments- Singular Value Decompositions
WT	Wavelet transform

List of publications

This thesis includes the following publications:

[P1] **A. Akhbardeh**, S. Junnila, M. Koivuluoma, T. Koivistoinen, V. Turjanmaa, T. Kööbi, and A. Värri, "Towards a Heart Disease Diagnosing System based on Force Sensitive chair's measurement, Biorthogonal Wavelets and Neural Network classifiers," Elsevier Journal of Engineering Applications of Artificial Intelligence (2006), http://dx.doi.org/10.1016/j.engappai.2006.07.005 (in press).

[P2] **A. Akhbardeh**, S. Junnila, M. Koivuluoma, and A. Värri, "Applying Novel Time-Frequency Moments Singular Value Decomposition (TFM-SVD) Method and Artificial Neural Networks for Ballistocardiography," EURASIP Journal on Advances in Signal Processing, accepted 10 September 2006, Vol. 2007, Hindawi Publishing Corporation, doi:10.1155/2007/60576 (in press).

[P3] **A. Akhbardeh**, and A Värri, "Applying Novel QuickLearn Algorithm for Pattern Recognition," 2006 IEEE International Conference on Systems, Man, and Cybernetics (SMC 2006), Taipei, Taiwan, 8-11 October 2006, pp. 851-855.

[P4] **A. Akhbardeh**, and A. Värri, "Novel supervised fuzzy adaptive resonance theory (SF-ART) neural network for pattern recognition," Proceedings of IEEE International Symposium on Intelligent Signal Processing, WISP 2005, University of Algarve, Portugal, 1-3 September 2005, pp. 149-154.

[P5] **A. Akhbardeh**, S. Junnila, T. Koivistoinen, and A. Värri, "Applying Novel Supervised Fuzzy Adaptive Resonance Theory (SF-ART) neural network, Biorthogonal wavelets for Ballistocardiogram diagnosis," 2006 IEEE Conference on Control Applications, , Munich, Germany, 4-6 October 2006, pp. 143-148.

[P6] **A. Akhbardeh**, S. Junnila, T. Koivistoinen, and A. Värri, "Applying Biorthogonal wavelets and a Novel QuickLearn Algorithm for an Intelligent Ballistocardiographic chair," 2006 IEEE Mountain Workshop on Adaptive and Learning Systems (SMCals'06), Utah State University, College of Engineering, Logan, U.S.A., 24–26 July 2006, pp. 42-47.

[P7] **A. Akhbardeh**, S. Junnila, M. Koivuluoma, T. Koivistoinen, and A. Värri, Evaluation of heart condition based on ballistocardiogram classification using compactly supported wavelet transforms and neural networks, Proceedings of the 2005 IEEE Conference on Control Applications, Toronto, Canada, 29-31 August 2005, pp. 843-848.

[P8] **A. Akhbardeh**, S. Junnila, T. Koivistoinen, T. Kööbi, and A. Värri, "Ballistocardiogram classification using a novel transform so-called AliMap and biorthogonal wavelets," Proceedings of IEEE International Symposium on Intelligent Signal Processing, WISP 2005, University of Algarve, Portugal, 1-3 September 2005, pp. 64-69.

[P9] S. Junnila, **A. Akhbardeh**, A. Värri, and T. Koivistoinen, "An EMFi-film sensor based ballistocardiographic chair: performance and cycle extraction method," Proceedings of 2005 IEEE Workshop on Signal Processing Systems, Design and Implementation, SIPS 2005, Athens, Greece, 2-4 November 2005, pp. 373-377.

[P10] S. Junnila, **A. Akhbardeh**, L. C Barna, Irek Defee, and A. Värri, "A Wireless Ballistocardiographic Chair," 28th Annual International Conference IEEE Engineering in Medicine and Biology Society, Aug.30-Sept.3, 2006, New York City, New York, USA, pp. 5932-5935.

Chapter 1

Introduction

1.1 Background

Machine learning is a challenging area of research because its ultimate goal is to reach the learning ability of humans. During the past several years, a large number of paradigms have been developed to simulate the learning and decision-making capabilities of the human brain. Artificial Neural Networks (ANNs) have been developed based on neuron-physiological models for learning input patterns, with neurons and nodes, synaptic connections and weights, etc [1,2]. The performance of any learning methods such as ANNs depends heavily on the availability of a representative set of training examples. In many practical applications, however, acquisition of such a representative data set is expensive and time consuming. Moreover, it is not always easy to prepare such a huge data set and in most cases data become available in small batches over a period of time. Other limitations of the existing techniques for incremental learning concern their ease of hardware and/or software implementation, their learning speed, and their suitability for real-time processing [3]. Multilayer perceptron (MLP), Radial Basis Function (RBF) neural networks, Kohonen networks, most self-organizing maps, probabilistic neural networks, and wavelet neural networks are some examples of popular ANN structures.

Indeed, developing a high performance as well as a high-speed classifier that uses minimum training data set with a small number of iterations for adjusting weights is difficult but essential. During the past several years, a large number of neural networks either supervised or unsupervised with new structures/ learning algorithms have been developed. Some of them are adaptive resonance theory, neural trees, modified Hopfield, and learner ++ (retraining incrementally using fuzzy "IF-THEN" rules) [4,5]. Most of the existing methods perform remarkably well when a sufficient amount of training data, enough learning cycles and the proper learning parameters are chosen. However, methods that do not deal with such important issues may potentially produce misleading information. Other limitations of the existing techniques concern their degree of success in the case of validation test with new data that are not presented yet to the network, their ease of hardware and/or software implementation, their stability across different patterns, and their suitability for learning incrementally from new data (real time learning) [1].

1.2 Author's Contributions

The focus of this dissertation is on pattern recognition to develop efficient learning algorithms and techniques to increase the performance of the current classifiers and improve their learning speed, e.g., by overcoming one of their major problems. A complete pattern recognition system consists of a sensor that gathers the observations to be classified or described; a feature extraction mechanism that computes numeric or symbolic information from the observations; and a classification or description scheme that does the actual job of classifying or describing observations, relying on the extracted features. The author's contributions in [P1-P10] are development of signal processing and pattern recognition parts which can be divided into three main categories as follows:

1. The signals to be classified, especially in biomedical field, are often considered non-stationary. In many cases both time and frequency domain properties of the signals are important. So, for feature extraction, we often need a time-frequency analysis method such as wavelet transforms. Wavelet transforms require no prior knowledge of the statistical distribution of data samples and the computational complexity and training time are reduced (see [P1] and [P5-P8]). Other feature extraction methods such as Time-Frequency Moments Singular Value Decompositions (TFM-SVD) and statistical features are studied in this dissertation (see [P2]). To evaluate them, these methods were applied to Ballistocardiogram data classification. The Ballistocardiogram (BCG) signal was first segmented using the Electrocardiogram (ECG) as the synchronization signal and later based only on the BCG signal itself (see [P9,P10]).
2. Use of neural networks for classifying the extracted features. For this purpose we applied two well-known neural networks, Multi-Layer Perceptrons (MLP) and Radial Basis Functions (RBF), to our example BCG (see [P1, P7]). The performance of neural networks depends heavily on the availability of training examples. However, the acquisition of a training data set is time consuming and difficult. Therefore, we developed some new learning algorithms with incremental learning ability (QuickLearn, CombilNet and its example SF-ART) which have better properties (lower computational load, higher speed in learning and easier to implement) than present learning algorithms. We compared our learning algorithms with MLP, RBF and Competitive Neural Trees (CNeT)-see [P3-P6].
3. A novel method called CombiMap transform was developed and tested on BCG data. This Map tries to eliminate redundant information without losing relevant information incorporated in spatial, time, and frequency domains. This map is able to extract most important information and quantify them using scalar values, mapping from a high dimensional space to one scalar value. It has three factors to decide which information of input data is most important. This transform can be used for pattern classification (see [P8]).

For all papers [P1-P10], the ideas and novel methods came up by the author. The simulation methodology was also realized by the author. Moreover, all the

simulations (except BCG segmentation in [P1], [P2] and [P7]), performance analyzes, and writing (except hardware as well as measurement related sections) were performed solely by the author. In [P8], Starr classification related sections were written by co-authors. Papers [P9] and [P10] are mainly focused on hardware and measurement issues that were written by co-authors. In these two papers ([P9-P10]), Blind segmentation related sections were only written by the author.

1.3 Thesis Outline

This dissertation is organized as follows. Chapters 2 and 3 focus on Ballistocardiogram, its use, the BCG Measurement and recording systems. Noise removal and segmentation methods are additional topics discussed in Chapter 3. Chapter 4 considers some feature extraction methods that are wavelet transforms, Singular value decomposition (SVD), statistical moments, and Time-Frequency Moments Singular Value Decomposition. In Chapters 5 and 6 we discuss some existing classification methods, novel methods for pattern recognition and results for well-known pattern recognition problems as well as for Ballistocardiogram Classification. The novel methods that are described in this dissertation are Combined Learning Network (CombilNet), Supervised Fuzzy Adaptive Resonance Theory (SF-ART); QuickLearn Algorithm, and CombiMap Transform. Finally, Chapter 7 draws some conclusions and possible future research directions.

Chapter 2

Ballistocardiography (BCG)

2.1 BCG signal

The Ballistocardiography is an old, noninvasive method that is used for recording the movements of the body synchronously with the heartbeat due to the left ventricular pump activity. Indeed, BCG reflects the mechanical activity of the heart. As the heart is pumping blood out of the heart, a reaction force opposite to the force moving the blood is generated [6,7]. This force, among with the forces generated by the respiration and disturbances caused by body movements, can be measured as BCG in the spine axis using some sensitive force sensor.

BCG is useful for learning cardiac cycle physiology because of its noninvasive application in humans, clear physiological and physiopathological analysis, and practical approach to considering cardiac output issues [8]. There are three types of BCG signals related to the displacement, velocity and acceleration sensors, but our main focus in this dissertation is a BCG signal generated by the displacement. An example of the ballistocardiogram (BCG) signal and its components for one cardiac cycle is shown in Fig. 2.1. If we compare BCG with the electrocardiogram (ECG), we can find that the ECG measures the electrical activity of the heart whilst the BCG measures the movements arising from myocardial contraction, the ejection and movement of blood from the heart [9]. The following figure (Fig. 2.2) shows an ECG

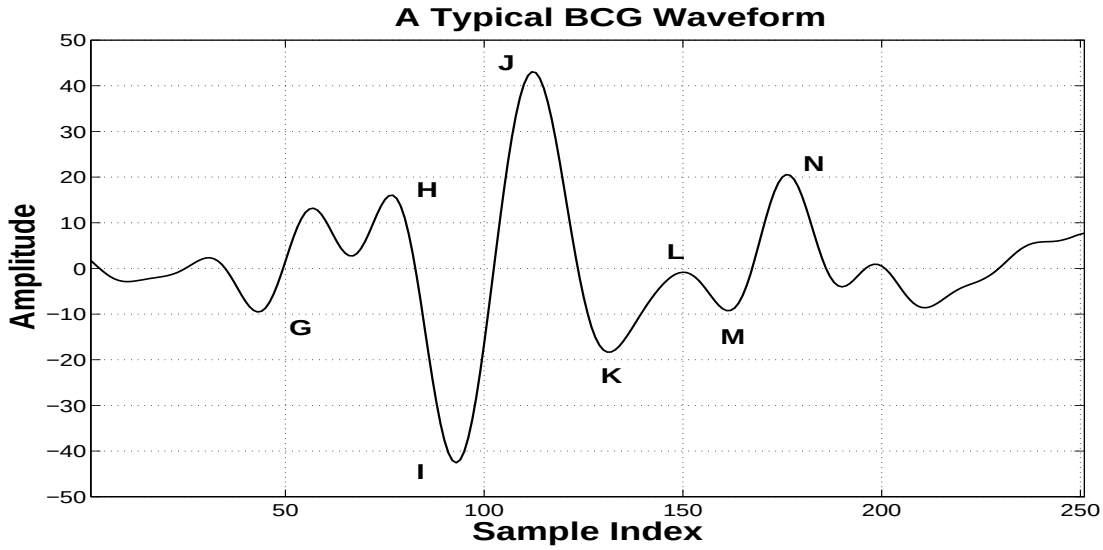


Fig. 2.1. Example of BCG signal including wave complexes called G, H, I, J, K, L, M, and N components.

and a BCG for a normal patient across more cardiac cycles.

Nihon Kohden made a table-type ballistocardiograph using a sensitive accelerometer in 1953 [10]. The operation of this device is explained by Harrison as follows: 'Blood from the heart is mainly ejected upwards along the ascending aorta. When pulling blood into the heart, the major motion is also along the axis parallel to the spine. Thus the major motion is longitudinal. For both ejecting and pulling blood, according to Newton's 3rd Law the force exerted on the blood by the heart is matched by an equal and opposite force on the body by the blood ($\vec{F}_{on\ blood} = -\vec{F}_{on\ body}$). If a patient is placed on a table with very low friction, then the force on the body causes the body and the table to move back and forth as the blood is being pumped. A sensitive accelerometer on the table measures its acceleration and one can compute the acceleration of the blood with: $m_{blood}\vec{a}_{blood} = -m_{blood+table} \cdot \vec{a}_{body+table}$. This is called the acceleration ballistocardiogram (BCG)' [11].

Starr classes

BCG signals can be divided into four different classes depending on the shape of the measured signals (Starr's classification) (Fig. 2.3). In class 1 all BCG complexes are normal in contour. In class 2 the majority of the complexes are normal, but one or two of the smaller complexes of each respiratory cycle are abnormal in contour. In

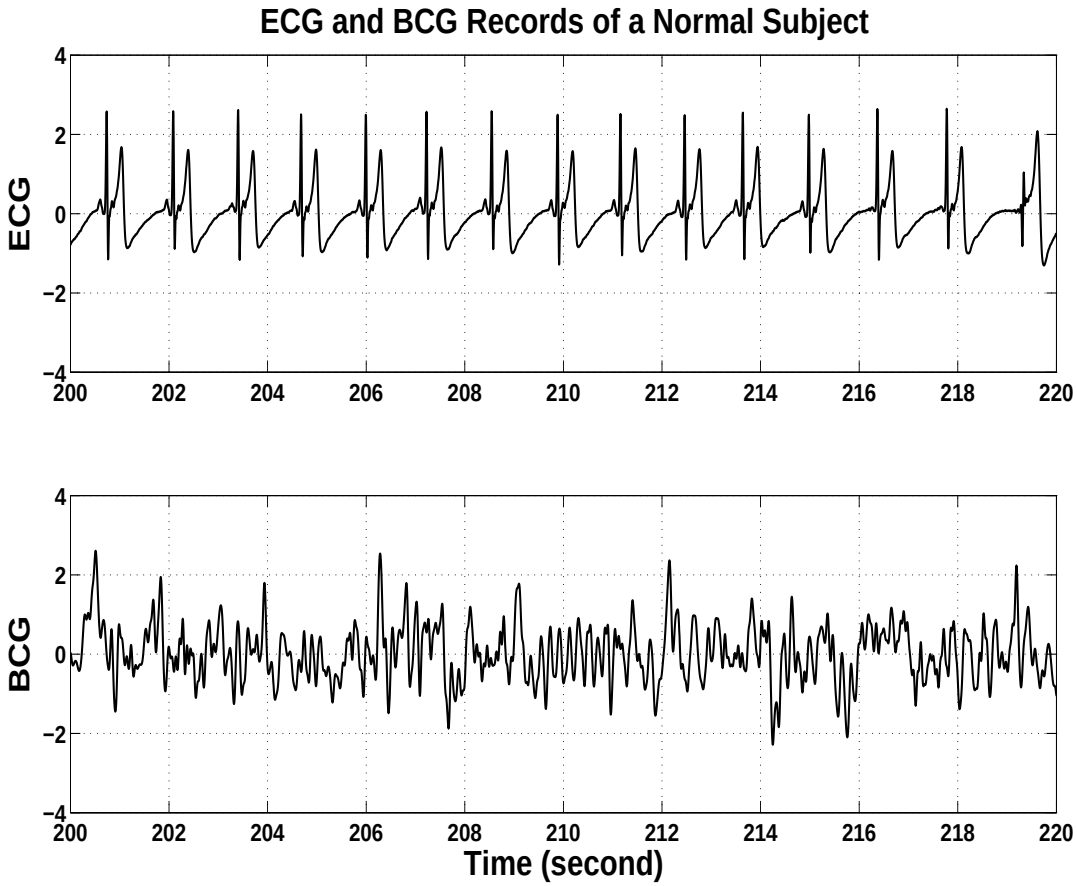


Fig. 2.2. Typical ECG & BCG records of a normal subject.

class 3 the majority of the complexes are abnormal in contour, usually only a few of the largest complexes of each respiratory cycle remaining normal. Finally, in class 4 there is such complete distortion that the waves can not be identified with confidence, and the typical BCG wave complexes shown in Fig. 2.1 can not be located without the assistance of a simultaneous ECG [12,13,14].

2.2 BCG applicability

The deflections of the BCG are attributed to forces generated by the heart when the blood moves, and the contour of the tracing is related to the shape of the cardiac ejection curve [15]. On the other hand, by comparing BCG records with the movement of the blood simultaneously, it was found that the contour of the BCG is mathematically related to the cardiac ejection curve.

Although the BCG has been known since the first half of the 20th century, it was not very useful as a diagnostic tool because the experimental errors in the measurement were large. Recently modern signal processing techniques have allowed these experimental errors to be greatly reduced, and BCG's are now better suited for the clinical applications. Evidence has been presented [15,16] that 'the size of the initial waves, 'I' and 'J', (see Fig. 2.1) is absolutely related to the cardiac output and that the form of the ballistic curve is determined by the shape of the curve of blood velocity in the great vessels. By these conceptions we can expect that, when the circulation is feeble, the BCG will be of low amplitude and, when the heart is weak, the form of the ballistic record will be altered'. In a follow-up study of Theorell et al. the increase in the I-J amplitude predicted a coronary death in patients who had experienced a myocardial infarction [17].

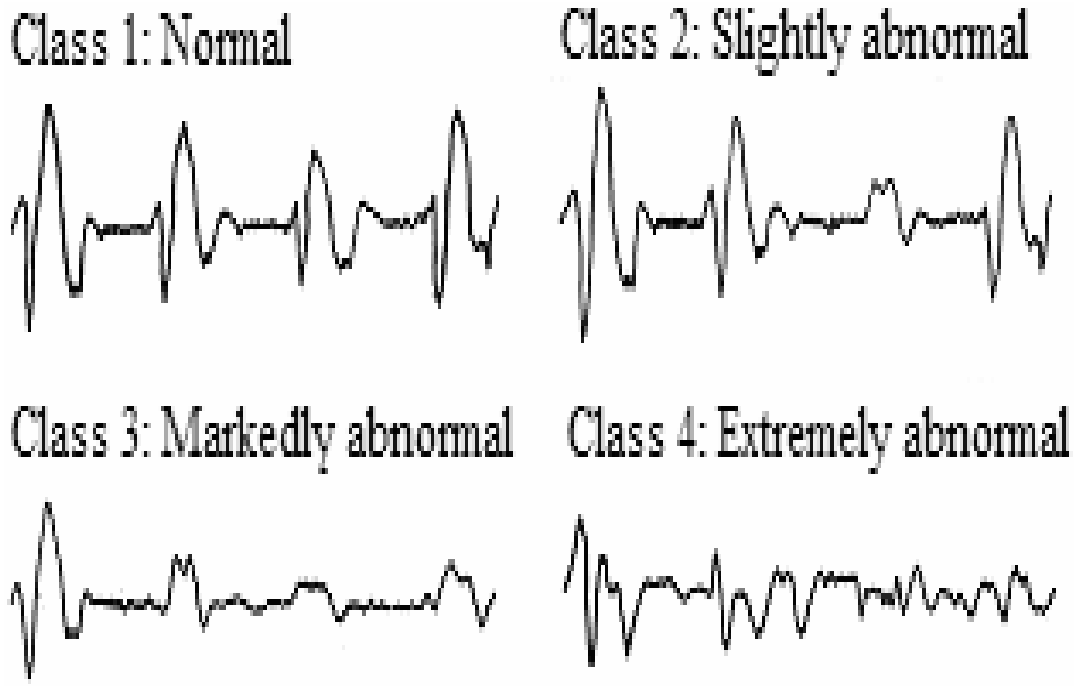


Fig. 2.3. Starr's classification. [12].

2.3 The BCG Measurements

The Ballistocardiograph (BCG) measurement was developed and used as a popular method before the 1940'is but since then other methods have replaced it, partly because the devices were difficult to construct.

In 1939 Starr et al. presented a Ballistocardiograph which is an adaptation of an old idea and consists of a table suspended from the ceiling on wires and braced to prevent motion in any but the longitudinal direction. Motion in this direction, opposed by a strong steel spring, is magnified about 8,000 times and photographed [15,16].

When a subject lies on the table he is not conscious of its motion but the records obtained are characteristic and reproducible. Figures 2.4 give some examples. 'The main forces producing this motion have been identified as the recoil from the ejection of blood from the heart, the impact of the blood striking the arch of the aorta and the curve of the pulmonary artery and, of less importance, the recoil from the systemic blood accelerated feet-ward when the aortic arch has been passed. The

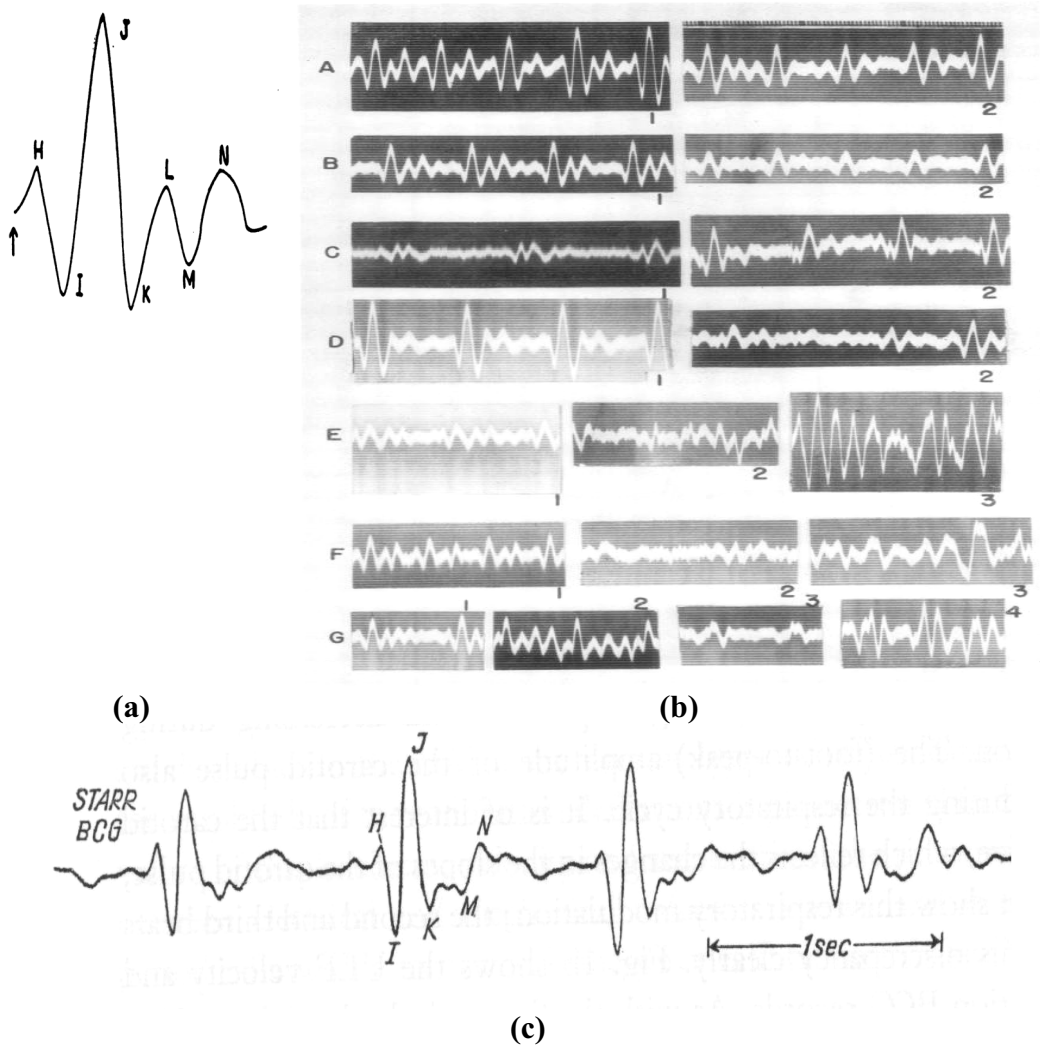


Fig. 2.4. Typical examples of waveforms measured by Starr; a) Diagram of normal ballistocardiogram with letters used to designate the waves [16]. b) Shows records obtained on 8 normal persons chosen to illustrate the extent of the normal variation [16]. c) Shows a long-time BCG record from [18].

resultant of these forces is not perfectly reproduced in the record, the difficulty being due to the physical properties of body tissues, which vibrate for a brief instant after receiving a single blow. These after vibrations warp the descending part of the record but they have a much smaller effect on the ascending part, and from this part reliable data can be secured' [16]. The relationship between the labels of BCG and the cardiac cycle is explained in [19].

In 1950, Starr et al. tried to determine exactly what a BCG measurement device actually measures (see Fig.2.5). 'While the subject lies on the ballistocardiograph a normal diastolic pressure is created and the heart's function is simulated by injecting fluid into the aorta and pulmonary artery, the amount injected at each instant being recorded. The resulting ballistocardiograms can be directly compared with many aspects of cardiac function. The amplitude of the ballistocardiogram measures the maximum force exerted by the heart in moving the blood and preliminary normal standards for this estimate of cardiac strength have been set up' [20].

Noodergraaf and H.Pollack collected some old BCG measurement systems that are basically based on electro-mechanical or optic- electro-mechanical methods in 1967 [18]. Some of them briefly described as follows:

- The completely isolated air ballistocardiograph: D.M. Cunningham et al. designed an isolated air ballistocardiograph which is shown in Fig. 2.6. It shows how it was difficult to record BCG in the past.

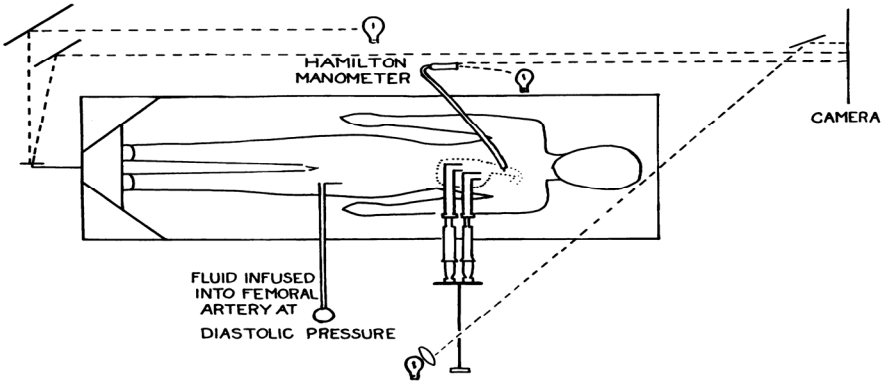


Fig. 2.5. Arrangement of apparatus developed by Starr in 1950 [20].

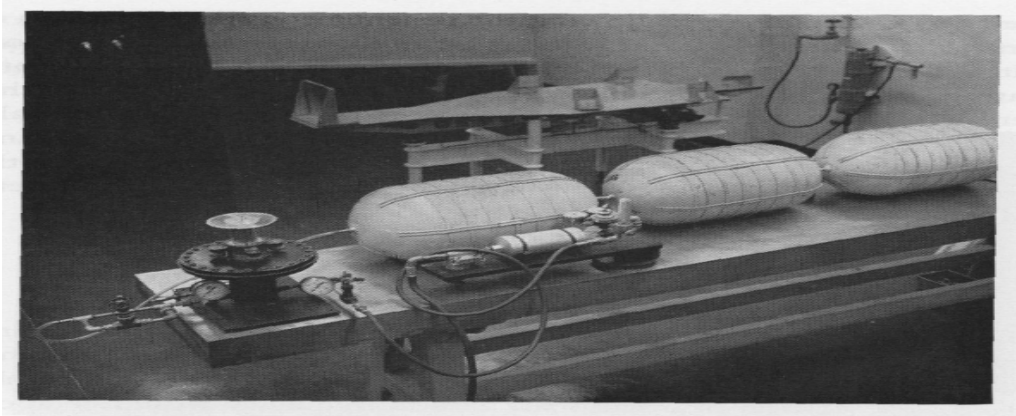


Fig. 2.6. Prototype of one of three supports of the completely isolated six degrees of freedom (6D) BCG showing current air thrust pad, air piston and spring, damping tank(s), and pressure regulator of the device developed by D.M. Cunningham et al. Image is from [18].

- A servo counterforce ballistocardiograph: Nyboer et al. developed a servo counterforce ballistocardiograph that is shown in Fig. 2.7. It is a method of recording force ballistics of respiration and circulatory pulses. 'It eliminates accessory damping and has an intrinsically flat frequency response of 0 to 16 Hz. The method is based on the principle that a counterforce applied to the platform to hold it at a zero reference position will be equal and opposite to the force applied to the platform by the subject' [18].

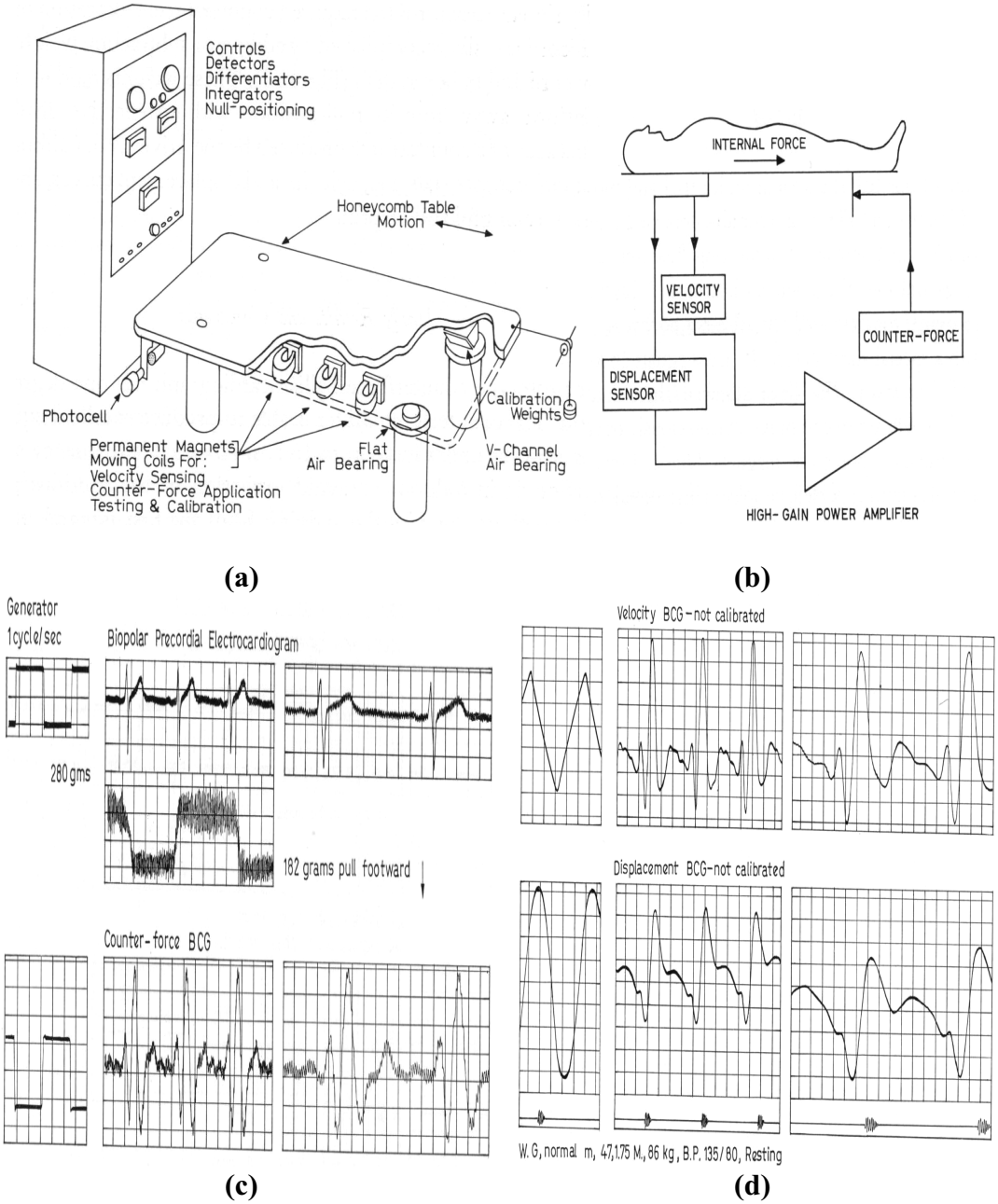
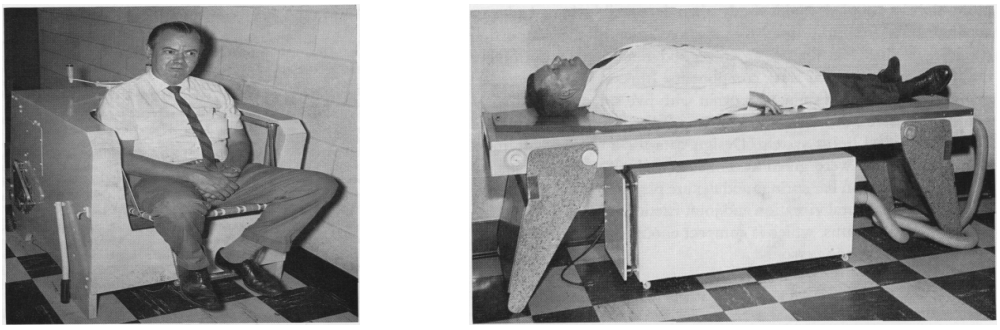


Fig. 2.7. Set up of the BCG measuring devices of Nyboer et al.; a) Mechanical concept of the servo counterforce BCG supported on air-bearings b) Electrical concept of the servo counterforce BCG. The system attempts to maintain the bed in its null-position by electromagnetic means, c), and d) during a held breath the counterforce BCG and its integrals are recorded electronically during the cardiac cycle as identified by simultaneous ECG of the subject under test. All images are from [18].

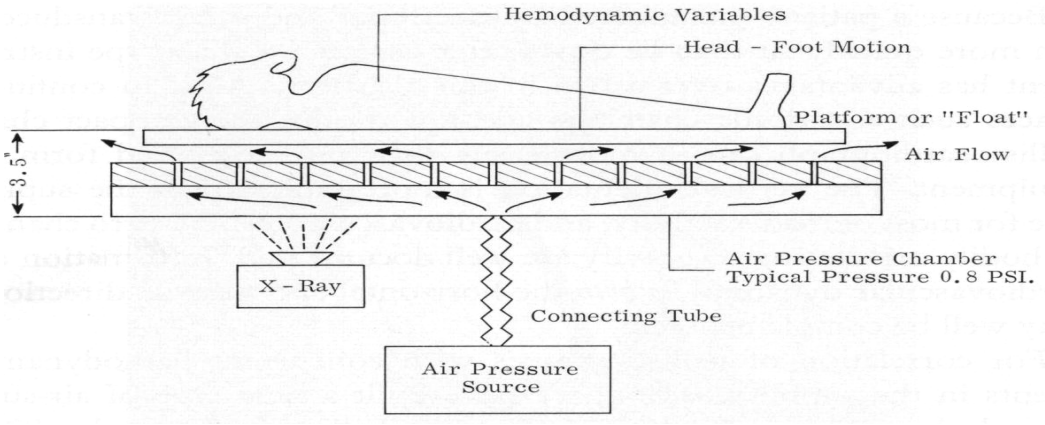
- Two forms of ultra-low frequency ballistocardiograph: W.K. Harrison et al., also developed two forms of ultra-low frequency ballistocardiograph, shown in Fig. 2.8. They designed a chair-type apparatus for recording vertical BCG, and an air-supported bed. 'Design features of the chair are a torsion-bar toggle mechanism and rugged but compliant flexure pivots to support the patient vertically on a 3.2 kilogram structure. The bed configuration, with a platform weight of about 4.5 kilogram, resembles a large air bearing in principle. Compressed air for its operation comes from a silenced industrial vacuum cleaner, and the small thickness of the apparatus makes its use on a fluoroscopy table possible'. Comparison of recordings from normal patients taken on each instrument showed that the ballistocardiographic IJ waves in the supine position are generally larger than those obtained in the sitting position [18].

Recently developed sensors offer new unobtrusive possibilities to measure BCG even at home. Thus, they are suitable for evaluation of the heart condition in any place. The classic BCG measurement, a freely suspended bed (or table), has been surpassed by more advanced and easier to use sensors, like the static-charge sensitive bed (SCSB) [21,22], shown in Fig. 2.9.

The SCSB is a sensitive movement sensor that is designed for long-term



(a) **(b)**



(c)

Fig. 2.8. Set up of the BCG measuring devices of W.K. Harrison et al.; a) Front view of the head-foot ultra-low frequency chair ballistocardiograph. The handle visible behind the patient adjusts the apparatus for patients of different weights b) the air-supported ultra-low frequency head-foot ballistocardiograph and power unit showing the removable legs for use outside the cardiac catheterization laboratory; c). Schematic diagram of the air-supported ballistocardiograph depicting the high pressure air plenum and flow limiting holes. All images are from [18].

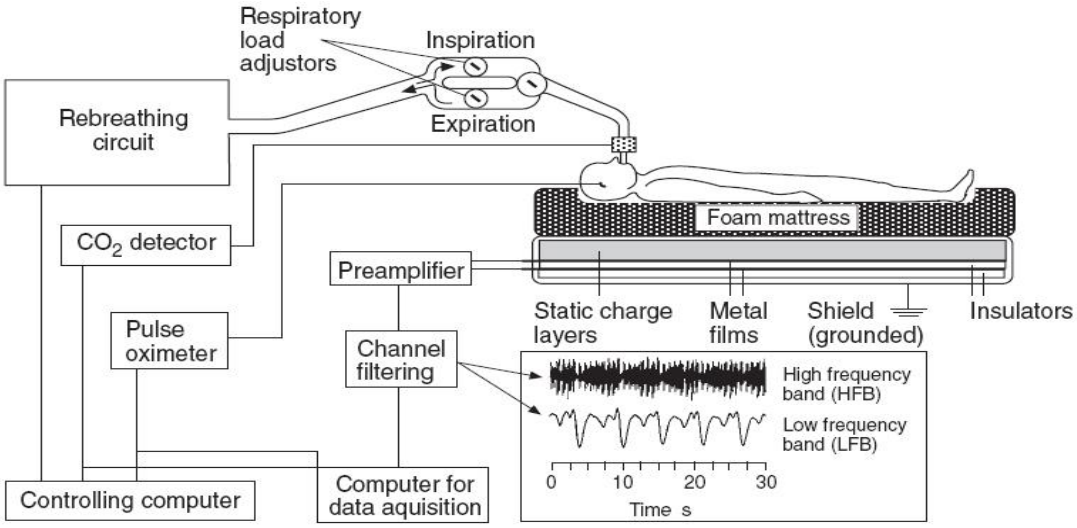


Fig. 2.9. Recording set-up with the static charge sensitive bed (SCSB) [21].

monitoring of gross body, respiratory and heart-related movements [23]. The records of these three different movements are contained within the composite signal generated by the SCSB, and are separated through analogue frequency filtering. The low frequency band (LFB) represents respiratory movements, whereas the high frequency band (HFB) reflects cardiac activity (the ballistocardiogram). In Finland and Sweden, the SCSB has been widely used since 1985 for the assessment of sleep-related breathing disorders [24, 25].

2.4 BCG signal analysis

BCG signal processing is a new field which requires much more research than what was done till now. During the past several years, a large number of general bio-signal classification methods have been developed, including single and multi channel template matching, principle component analysis, amplitude separation, Fourier analysis, linear filtering autoregressive modeling, neural networks, and maximum likelihood. Some of the methods used for BCG feature extraction (frequency domain analysis using FFT, Principal Component Analysis and wavelet transform) and classification (neural network) have been used by Xinsheng Yu [26,27,28], but more works are still necessary for the BCG analysis.

Most of the existing methods for biomedical signal processing perform very well when the problems of motion artifacts, signal latency and non-linear disturbance (electrical drifting of electronic devices) are not considered. However, methods that do not deal with such important issues may potentially produce misleading information about the patient. Other limitations of the existing techniques concern their degree of success in the case of special situation of subject such as stress, their ease of hardware and/or software implementation, their portability across platforms, and their suitability for real-time processing [29,30,31]. Thus, in this study we tried to apply different feature extraction and classifiers for the BCG classification and overcome the above mentioned problems.

Chapter 3

BCG Measurements in ProHeMon project

3.1 ProHeMon project

BCG permits the identification of patients with abnormal circulations and provides evidence of cardiac health or disease of a type that has no counterpart in the ordinary clinical tests. For this purpose it was necessary to do research on normal and abnormal hearts. To this end over 100 subjects, both healthy and unhealthy persons from 20 to 70 years of age were recorded/examined using the designed measurement system during the ProHeMon project (Proactive Health Monitoring, sponsored by the Academy of Finland 2003-2005). Using existing and developed signal processing and pattern recognition techniques that we will discuss in the next chapters, we tried to find Characteristic BCG patterns for healthy and unhealthy people.

3.2 The designed BCG measurement device

A bed size sensor requires a lot of space to operate, which has motivated us to develop a chair based sensor system. New sensor materials, such as the EMFi sensor [32] offer new possibilities to implement BCG devices easily and the developments of signal processing methods enable far more efficient analysis of the signals than in the past. The developed EMFi-sensor based chair can be made very light and easily transportable, and using wireless data transmission even indistinguishable from a normal chair. No electrodes are needed, and the patient can even be measured fully clothed, although very thick clothing does dampen the signal.

The recording system applied in this study is based on force sensitive chair's measurement developed by the ProHeMon research team located in Tampere University Hospital and Tampere University of Technology-Institute of Signal Processing (see Fig. 3.1) [33,34].

The measurement chair is a normal office chair with separate plywood seat and backrest. The EMFi-sensors are fitted on the seat and backrest under very thin textile

upholstery (Fig. 3.2). Changes in the thickness of the electromechanical film produce a charge change, which can be measured from the film terminals using a pre (charge)

Fig. 3.1. Setup for recording BCG developed by ProHeMon research team.

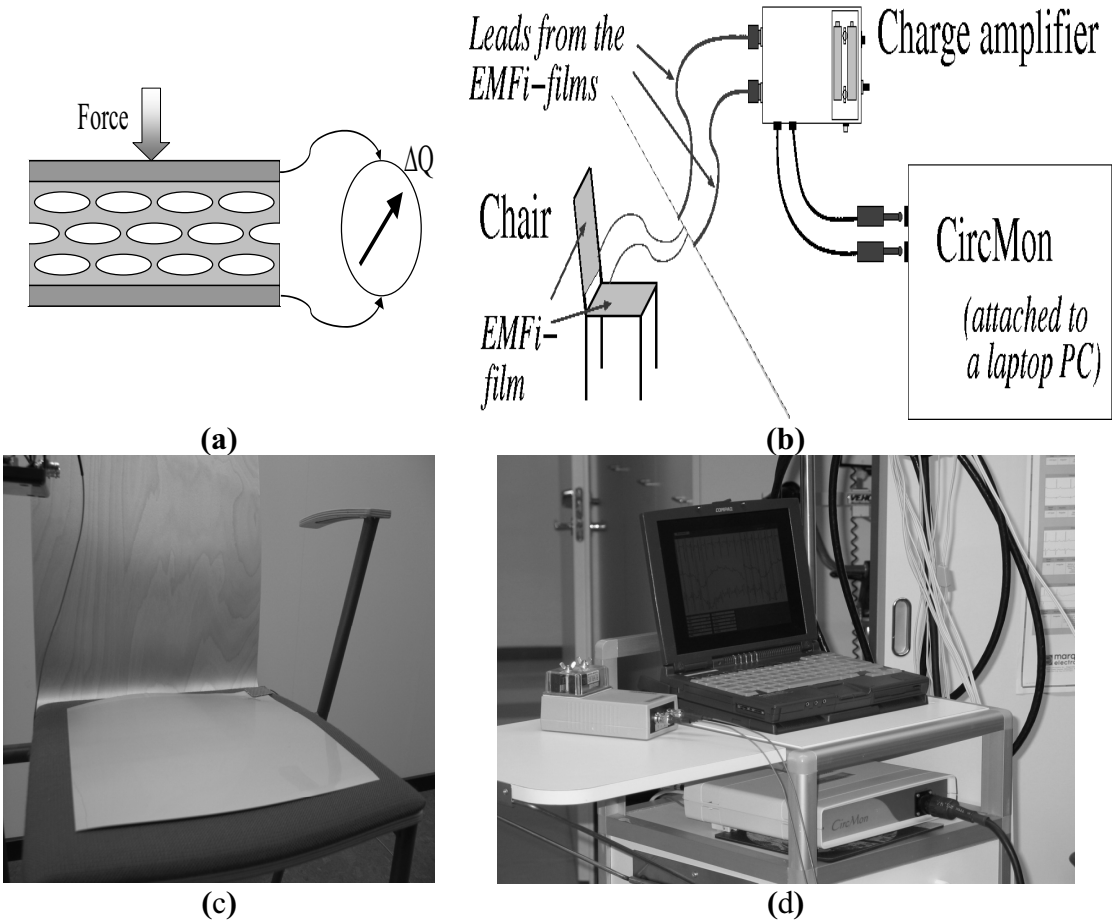


Fig. 3.2 Set up of the measuring devices; a) EMFi™-Sensor operation, b) Sensor-Charge amplifier-data acquisition interface schematic c) Sensor film and its size compared with chair surface, and d) Charge amplifier-CircMon-PC interfaces [P1].

amplifier. The charge amplifier is implemented as a separate battery powered pre-amplifier device to convert and amplify the weak charge change signal to a voltage signal, which is then low-pass filtered with Frequency cut-of 240 Hz. The output of the pre-amplifier is sent to a data acquisition unit, so-called CircMon developed by Jr Medical Ltd, Tallinn/Estonia to amplify the signal again (post-amplifier) and then sample and record it with the frequency of 200 samples/second. During the BCG recording, ECG signal is recorded simultaneously from the chest of subject at the same sampling rate by the CircMon device. Figure 3.2 shows also chair-sensor-charge amplifier-CircMon-PC interfaces to record, monitor, and store BCG/ECG-data.

An example of the recorded BCG signal is shown in Fig. 3.3. EMFi-film sensor generates a movement related signal consisting of components attributable to BCG, respiration, and body movement (Fig 3.3).

3.3 Noise removal

Raw BCG is difficult to analyze by medical doctors, and its use in medical diagnosis has never been very common. With the development of computer processing power and signal processing algorithms, new interest in the BCG has risen. Using band pass filtering with different bands we can extract the respiration and BCG signals (Fig.3.4).

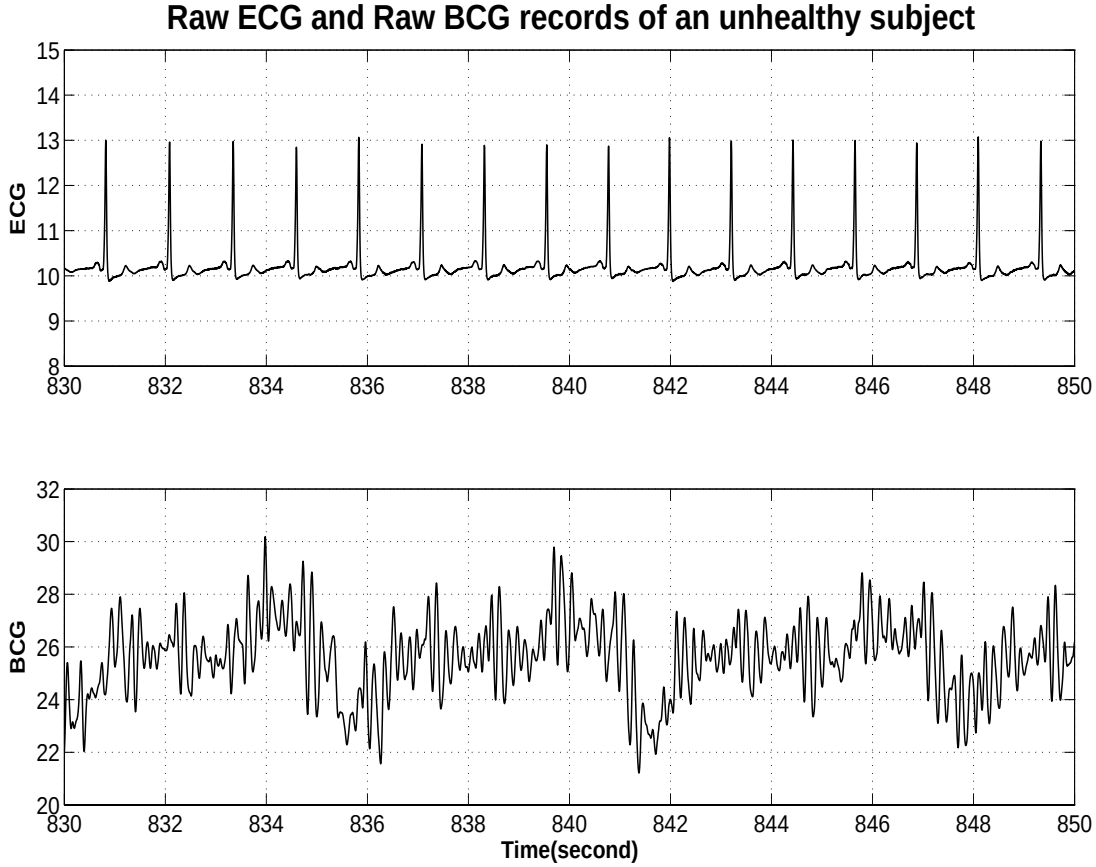


Fig. 3.3. Typical raw ECG & BCG records of an unhealthy subject.

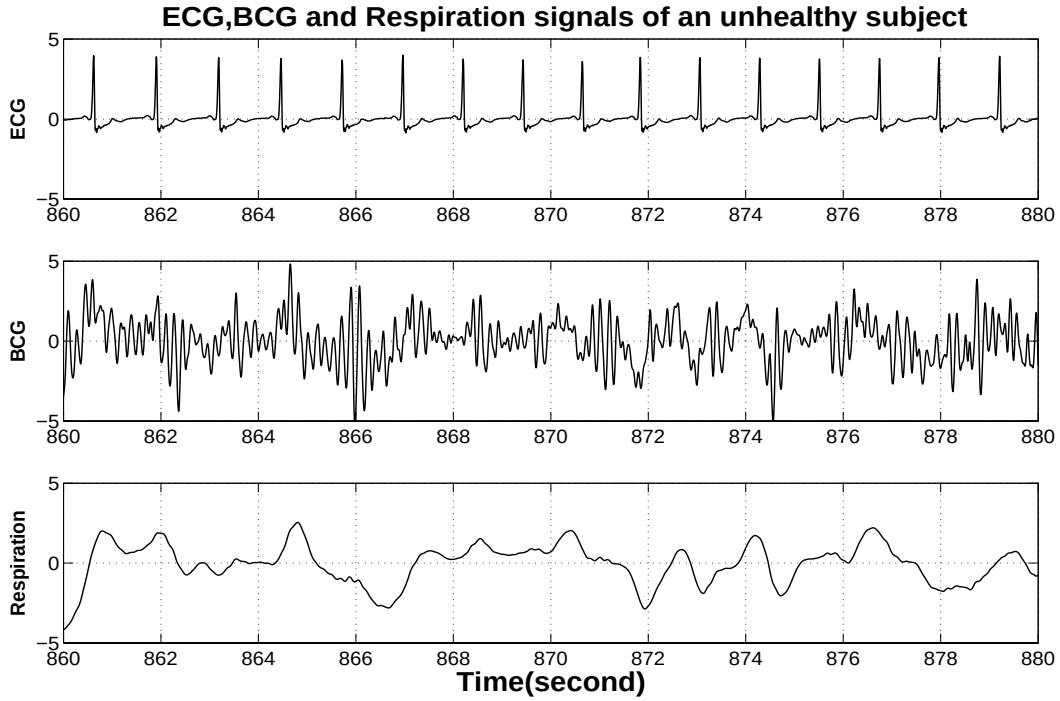


Fig. 3.4. Typical ECG, BCG and respiration signals of an unhealthy subject. Signals are filtered using band pass filters [1,40] Hz for ECG, [2,20] Hz for BCG, and [0.1 1] Hz for the respiration signal.

However, body movements during recording destroy some BCG cycles (increasing high amplitude components to the signal). These parts of the recorded BCG signal are useless and must be eliminated by amplitude thresholds for accounting only valid parts of signal. This method is useful because body movements usually cause bigger signal changes compared to normal cardiac activities. A more rigid chair than used in this study might improve the quality of the signal slightly. The main reason for the artifacts is, however, the movement of the subject. It is often difficult to find 60s long periods of BCG without movement artifacts but periods of 30s can be found of BCG easily when the subject has been asked to sit still (Barna l., personal communication, 2006).

3.4 BCG Segmentation Methods

To extract BCG waveforms and synchronization with cardiac cycle, the R-Component of the ECG signal can be used to identify each BCG cycle using amplitude/threshold separation method and BCG waveforms can be extracted into a unique 250 points window which is long enough to show a complete cardiac cycle and all BCG components. The advantage of this method is the extraction of BCG cycles with high accuracy and their synchronization with cardiac cycles [P1, P2].

However, several methods have been developed for signal segmentation without using any other synchronization signal. One of the simplest examples is auto-correlation, which for an almost periodic signal peaks at its period [35,36]. It uses a fixed reference window and a moving test window with the same width. Auto-correlation functions must be computed for both windows. The autocorrelation

function for the second window is a time variable because it must sweep the whole signal. Then either a special auto-correlation distance measure (d(t)) between two auto-correlation functions or cross-correlation between two windows must be calculated. If the d(t) measure is used, the cycles match the best when the distance measure reaches a local minimum. But, if a cross-correlation between two windows is used, the cycles match the best when cross-correlation reaches its local maximum which should be close to one. Spectral error measure (SEM) [37], Appel's generalized likelihood ratio (GLR) [38], the nonlinear energy operator (NLEO) [39], forward/backward predictors based [40], Wavelet-Based [41], and time-varying AR(p) model (TVAR) [42,43] are other examples of existing segmentation techniques which especially applied for the EEG segmentation. Värri in [36] and Glavinovitch et al. in [41] collected some of EEG segmentation methods which can be also applied for BCG segmentation.

As can be seen in Fig. 2.3, BCG cycles can be time-varying and they differ a lot for healthy and unhealthy people, the auto-correlation method with a fixed reference window generates error and might be not robust. Other above mentioned segmentation methods could be even more robust but definitely they need more computational load than our first method called blind segmentation method. It uses an absolute value of the respiration signal (obtained by filtering the BCG signal) as a synchronization signal and runs a non-mathematical rule based algorithm for BCG segmentation [P9]. The problem with using respiration signal is the loss of some of the BCG cycles because of the time period of the respiration signal.

To overcome this problem, we suggested a second version of the blind segmentation method in [P10] extracting a coarse BCG signal using a narrow band pass filter with 1 and 2 Hz corner frequencies. In the same way as with the respiration signal, we used absolute values of this coarse BCG signal and its peaks within lower and upper amplitude thresholds as synchronization points, eliminating peaks from the defined range. Based on our experience, peaks from this range are not related to BCG cycles, being background noises or motion artifacts.

Uniform windows with a length of 250 samples (1.2 seconds) were used to extract individual BCG cycles. The synchronization points computed from the coarse BCG signal are used to find the central points of these windows and 125 samples before and 124 samples after these central points are taken to create BCG cycles with the same lengths. The cycle lengths are not adaptive to heart rate (HR), as we tried to extract the BCG cycles without the use of ECG (which could have been easily used to estimate the HR quite accurately), and the estimation of R using only BCG was not accurate enough. Fig. 3.5 shows a sample BCG signal and the calculated coarse BCG signal.

Table 3.1 shows the difference between the numbers of extracted cycles using the above-mentioned two different blind segmentation methods. Cycles were extracted from five-minute BCG recordings of three subjects representing different patient groups. As can be seen, the new coarse signal based method is able to extract more cycles than the respiration signal based. However, as the average heart rate of the subjects was around 60-75 bpm in all the recordings (about 300-375 beats per 5 minute recording), it can be seen that the new algorithm detects about three synchronization points per every two cardiac cycles.

To verify this, we performed a one-minute beat-to-beat comparison against recorded reference ECG. The results can be seen in Table 3.2. For a typical young healthy man the method extracted 110 BCG cycles over one minute. The correct number of cardiac cycles was 75 of which the method extracted 74 cycles correctly. The other 35 cycles were repeated, redundant, and the same as previously extracted BCG cycles. Because the BCG signals for old people are mostly weaker and more re-shaped, the identification of their BCG cycles is more difficult than for BCG signals of young people. So, as can be seen in Table 3.2, the blind BCG segmentation method makes more errors for old people than for young people. We also found one undetected cycle, which was not detected because of a large motion artifact that destroyed a BCG cycle and caused the method to ignore it. Fig. 3.6 shows six extracted BCG cycles of an old man with a past cardiac infarct, extracted using the coarse BCG method.

Overall, the blind segmentation method is an automated method for period segmentation based on peak detection. It is a simple and efficient method with a low computational load which depends on the localization of single points of the signal (peaks). If strong noises such as motion artifacts deform signal peaks, this method is able to detect them using a simple threshold and eliminates them from further analyses. On the other hand the method only segmented BCG cycles in which their (absolute) peaks were in the normal range. The strong as well as weak peaks were ignored.

As explained in [P10] and above, this method gives some redundant BCG cycles because of unrelated local maximums but it is not a critical problem for pattern recognition in this case. To prove this, in [P1-P8] we applied both R-spike based and blind segmentation methods for the BCG classification and we didn't find a large difference between the two methods for the BCG classification. That's why we didn't apply other existing methods for the BCG segmentation.

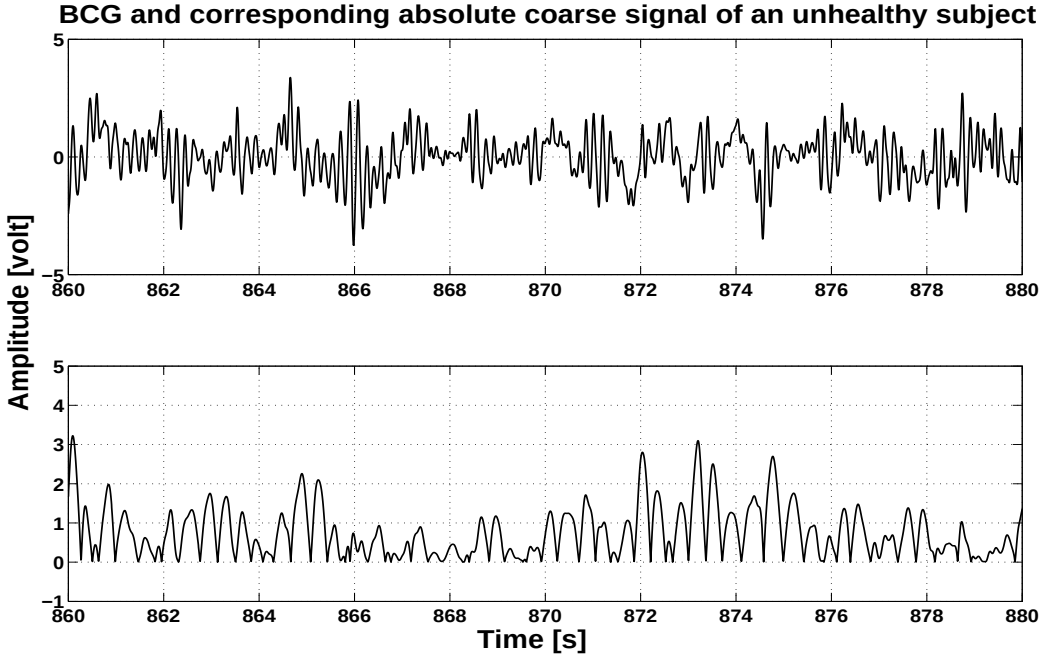


Fig. 3.5. Typical BCG and corresponding absolute values of coarse signal for an old man with a past cardiac infarct.

TABLE 3.1 Blind Segmentation Results for five-minute BCG recordings of three subjects [P10].

Subject	Number of cardiac cycles extracted from BCG	
	Using respiration signal	Using coarse signal
Young healthy man	215	582
Old healthy man	271	571
Old man with past cardiac infarction	259	515

TABLE 3.2 Beat-by-Beat Evaluation of Blind Segmentation Methods for one-minute BCG recordings of three subjects [P10].

	Number of ECG Cycles (Heart rate)	Number of cardiac cycles extracted from BCG			
		Total	Correct	Redundant	Missed
Young healthy man		110	74	35	1
Old healthy man		130	67	63	0
Old man with past cardiac infarction		142	61	81	0

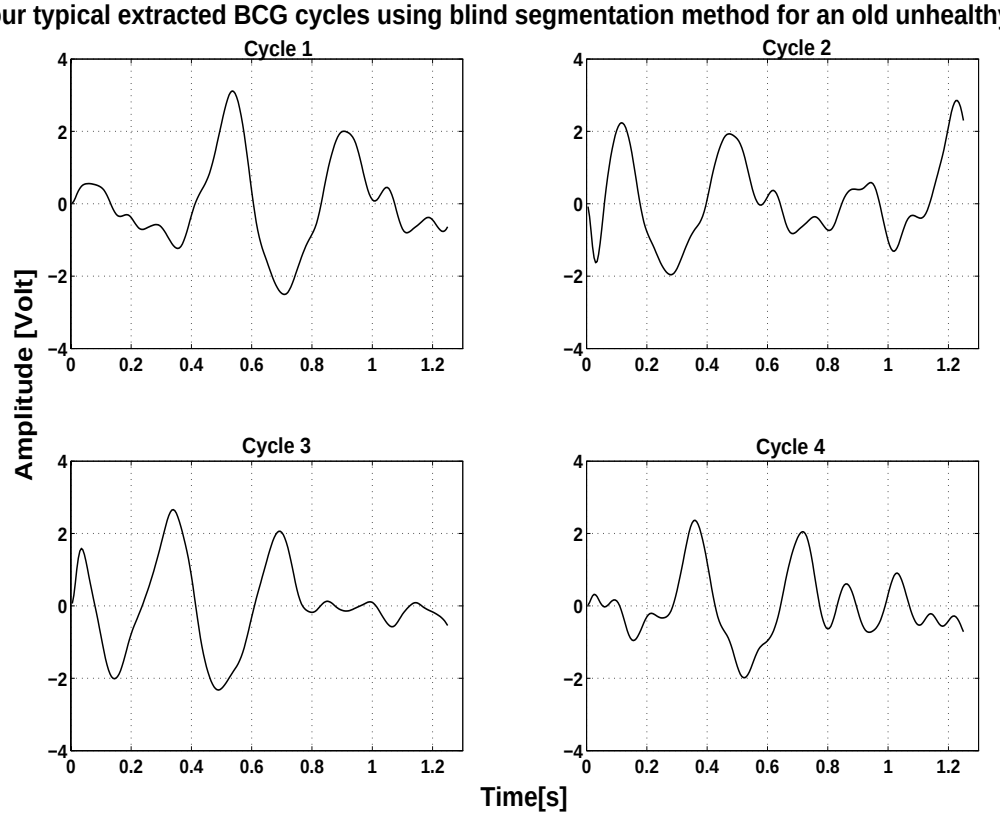


Fig. 3.6. Four examples of BCG cycles of an old man with past cardiac infarct extracted using the blind segmentation method.

Chapter 4

Feature extraction and dimension reduction

4.1 Introduction

Feature extraction is a special form of dimensionality reduction and is in the area of signal/image processing also connected with pattern recognition. As can be seen in Fig. 4.1, conventional pattern recognition systems have three components: signal segmentation, feature analysis and pattern classification. As explained in the previous chapter, any long-time signal such as BCG must be cleaned first from noise and then it must be segmented. Noise removal can be done using a simple band-pass filter or even adaptive noise canceling methods. Indeed, certain badly corrupted signal segments simply need to be rejected. Here, feature analysis is achieved in two steps: parameter extraction step and feature extraction step. In the parameter extraction step, information relevant for pattern classification is extracted from the input data in the form of a parameter vector. In the feature extraction step, the parameter vector is transformed to a feature vector. Feature extraction can be conducted independently or jointly with either parameter extraction or classification. The wavelet transforms (WT) [44,45] and Singular Value Decomposition (SVD) [46], which we used in this study, are two popular feature extraction methods. Both of them extract features by projecting the input vectors into a new feature space. Finally, the signal classification can be done using classifiers such as neural networks which are explained in the next chapter. For BCG analysis, the subjects were classified to N=3 classes which were young healthy, old healthy and old unhealthy.

To process a signal there are several time or frequency analysis methods and transforms. Indeed, methods such as Fourier analysis have been applied for stationary signals which are constant in their statistical properties over time. They work well if the signal is composed of some stationary components. However, any sudden change in time is expanded over the whole spectrum of frequencies in the Fourier transform of the signal. So, for non-stationary random processes if we use joint time-frequency analyses the signal processing results will improve. As can be seen in Fig. 4.1, the feature extraction must be done after signal segmentation (non-overlapping windows). So, we can assume that windowed signals (cycles) are stationary. Although methods such as short-time Fourier Transform (STFT) and the WT are linear, they take advantage of the local stationarity assumption for the signal which gives us more freedom than the assumption of overall stationarity. Therefore, we can apply them for non-stationary signals such as biological signals [47,48].

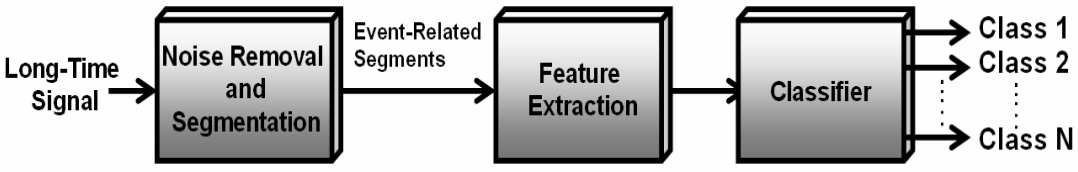


Fig. 4.1. Block diagram of a conventional pattern recognition system to classify input signal into N classes.

STFT has many properties of the Fourier transform, but its performance depends critically on the choice of the window and its type. However, WT doesn't use any window and tries to decompose a signal using some wavelet bases which are explained in the next section. WT has a higher time-frequency resolution and it is more robust than STFT that applies Fourier transform for sliding windowed signal [47]. So, in this study we selected Wavelet transforms that are a form of high-resolution time-frequency analysis and have less sensitivity to latency and non-linear disturbances. Moreover, the wavelet transform requires no prior knowledge of the statistical distribution of data samples and the computational complexity is reduced. We also developed a novel feature extraction method called Time-Frequency Singular Value Decompositions (TFM-SVD) the performance of which is comparable with WT [P2].

4.2 Wavelet transforms

The continuous wavelet transform of a square- integrable function $f(t)$ is defined as

$$wf(s,t) = \int_{-\infty}^{+\infty} f(t) \frac{1}{\sqrt{s}} \psi\left(\frac{t-\tau}{s}\right) d\tau \qquad (4.1)$$

Where s and t are the scale (or frequency) and time variable, respectively. The function $\psi(t)$, called wavelet, must satisfy the admissibility condition, i.e. it must be a zero-mean, square-integrable function. In practical applications the parameters s and t must be discretized. The simplest way is dyadic (2^j). By using it, the wavelet transform can be presented as:

$$wf[n,2^j] = \sum_{m=0}^{N-1} f[m]\psi_j[m-n] = f[n] * \psi_j[n] \qquad (4.2)$$

Where $\psi_j[n] = \psi\left(n/2^j\right)$ and $f[n]$ is a sequence with length N, and the sign * represents circular convolution. If wavelet transforms of $f(t)$ and $f(t-\tau)$ are the same, then it is called shift-invariant wavelet transform [44,45].

Fast wavelet transform (FWT)

First, the scale parameter of the continuous wavelet transform must be discretized around a k sequence $\{k^j\}_{j\in z}$. For instance, if we use k=2, the transform will be called dyadic wavelet. To implement a fast computing transform with shift invariant property, an algorithm called Mallat is presented in [41,42,45]. Suppose $a_0[n]$ is a

sequence of input signal samples. For $j>0$, we denote: $a_j[n] = \int_{-\infty}^{+\infty} f(t)\psi(n-t)dt$.

The dyadic wavelet coefficients are computed for $j>0$ over the integer grid

$d_j(n) = wf(n,k^j) = \langle f(t), \psi_2(t-n) \rangle$

For any filter $x[n]$, we denote by $x_j[n]$ the filters obtained by inserting $2^j - 1$ zeros between samples of $x[n]$ with Fourier transform equal to $X(k^j\omega)$. Inserting zeros in the samples of $x[n]$ creates holes. Now, let $\overline{x}_j[n] = x_j[-n]$. By using the following equation, we can compute a dyadic wavelet transform and its inverse:

$a_{j+1}[n] = a_j[n] * \overline{h}_i[n], \quad g_{j+1}[n] = a_j[n] * \overline{g}[n]$.

Filter $h[n]$, a so-called conjugate mirror filter is a low-pass filter and hence only low frequency components can pass through it; on the other hand $g[n]$ is a high-pass filter. The dyadic wavelet representation of signal a_0 is defined as the set of wavelet coefficients up to a scale k^J plus the remaining low-frequency information $a_J : \left[\{d_j\}_{1 \le j \le J}, a_J \right]$. Fig. 6 in [P1] shows block diagram of such operations, where $J = \log_2 N$ and $\tilde{x}[n]$ is the duality of $x[n]$.

Choosing proper wavelet bases

Choosing proper wavelet bases (ψ) is very important. Most application of wavelet bases exploits their ability to efficiently approximate particular classes of functions with few non-zero wavelet coefficients. This is true not only for data compression but also for noise removal and fast computation. The design of ψ must therefore be optimized to produce a maximum number of wavelet coefficients that are close to zero. A signal $f(t)$ has few non-negligible wavelet coefficients if most of the fine-scale (high resolution) wavelet coefficients are small. This depends mostly on the regularity of *f,* number of vanishing moments of ψ and size of its support. To construct an appropriate wavelet, these properties must be considered:

- Vanishing moments: as definition, ψ has P vanishing points if $\int_{-\infty}^{+\infty} t^k \psi(t)dt = 0 \quad for \quad 0 \le k < p$. It has been proved that if f is regular and ψ has enough vanishing moments then the wavelet coefficients are small at fine scale 2^j.
- Size of support: as definition, support size is the bandwidth for which ψ has considerable sensitivity. It has been proved that we must reduce support size of ψ for reducing the number of high amplitude coefficients. However, it has been found that the support size of a function and the number of vanishing moments are independent.
- Regularity: the regularity of ψ has mostly a cosmetic influence on the error introduced by thresholding or quantizing the wavelet coefficients. When reconstructing a signal from its wavelet coefficients: $f = \sum_{j=-\infty}^{+\infty} \sum_{n=-\infty}^{+\infty} \langle f, \psi_{j,n} \rangle \psi_{j,n}$,

an error ε added to a coefficient $\langle f, \psi_{j,n} \rangle$ will add the wavelet component $\varepsilon\psi_{j,n}$ to the reconstructed signal. If ψ is smooth, then $\varepsilon\psi_{j,n}$ is a smooth error. For image/signal processing applications, a smooth error is often less visible than an irregular error, even though they have the same energy [P1], [44].

Daubechies compactly supported Wavelets

One of the compactly supported wavelets is daubechies. These bases are orthogonal wavelets and have less information redundancy than other wavelet transforms. Daubechies wavelets have a support of minimum size for any given number P of vanishing moments. Daubechies wavelets are explained in more detail in [P7].

Biorthogonal wavelets

Other interesting wavelets are Biorthogonal Spline wavelets. Biorthogonality means that wavelet bases are bilinear as well as orthogonal. There are several parameters that affect the wavelet characteristics and behavior. In real applications such as feature extraction, we would like to decrease the number of the high amplitude values (singularities) of the wavelet coefficients as well as to increase the number of around-zero coefficients. Because of the tradeoffs in parameter choices we cannot completely fulfill both. If a wavelet gives us only one or two factors to control its behavior, it is recommended to choose it for our application because of this benefit. In biorthogonal wavelet, it is possible to control its behavior using two factors (p and $\tilde{p}$). The nonlinearity increases by p. The number of around-zero wavelet coefficients as well as number of high amplitudes increases by $\tilde{p}$ (tradeoffs). Thus, we must be aware about singularities of signal and generating high amplitude coefficients while increasing $\tilde{p}$ (only increasing $\tilde{p}$ till generating enough around-zero wavelet coefficients). In [P1], we explained Biorthogonal wavelets in more detail.

4.3 Singular value decomposition

Singular value decomposition is used because it captures the essential information of a matrix, somewhat similarly as the Eigen Values do. A singular value and the corresponding singular vectors of a rectangular matrix A are a scalar σ and a pair of vectors u and v that satisfy

$$A\, v = \sigma u$$
$$A^T u = \sigma v$$

With all singular values placed on the diagonal of a diagonal matrix and the corresponding singular vectors forming the columns of two orthogonal matrices U and V, we have

$$AV = U\sum$$
$$A^T U = V\sum$$

Since U and V are orthogonal, this becomes the singular value decomposition

$$A = U\sum V^T$$

The full singular value decomposition of an $m \times n$ matrix involves an $m \times m$ U, an m-by-n $\sum$, and an n-by-n V. In other words, U and V are both square matrices. If A has many more rows than columns, the resulting U can be quite large, but most of its columns are multiplied by zeros in $\sum$. In this situation, the economy-sized decomposition saves both time and storage by producing a $m \times n$ U, a $n \times n$ $\sum$ and the same V [P2, 46].

The singular value decomposition (SVD) is an appropriate tool for analyzing a mapping from one vector space into another vector space, possibly with a different dimension. If the matrix under analysis is square, symmetric, and positive definite, then its Eigen Value and singular value decompositions are the same. In particular, the singular value decomposition of a real matrix is always real, but the Eigen Value decomposition of a real, non-symmetric matrix might be complex [46]. However, if we use it to find SVs of a $m \times 1$ or $1 \times m$ array with elements representing samples of a signal, it will return only one singular value that is not enough to express all parts of a signal. So we cannot apply standard SVD for feature extraction of a signal. To overcome this problem we will introduce by the end of this chapter novel Time-Frequency Singular Value Decompositions (TFM-SVD) method [P2]. It applies the standard SVD for a 2x2 feature matrix (M) defined using statistical features of the signal and its spectrum which does not mean physically the signal itself. Indeed, TFM-SVD uses the mathematical concepts of the SVD to represent the feature matrix (M) by four scalar singular values and it does not have any physical correspondence compared to the philosophy of using SVD in the control and automation field.

However, SVD has other applications in digital signal processing such as noise reduction using subspace methods (TSVD: Truncated SVD noise reduction). Hansen in [49] explained a few SVD applications in scientific computing, signal processing, automatic control, and many other areas. 'The key idea of using signal subspaces and the SVD for noise removal is to define a Hankel matrix (A) by the vector signal matrix (S) which includes the noisy signal, compute the SVD, and then ignore small singular values of matrix A which mainly show the noise and the rank-k matrix A_k gives a filtered signal with less noise. The value of k must be giving the enough dimension of the signal subspace to catch the pure signal. By increasing k, the signal subspace will be able to catch a filtered signal with lower noise' [49,50].

4.4 Statistical features

Four popular statistical moments of time-series x[n] which were used in this study are:

Mean=E[x] = μ,

Variance_t=E [$(x - \mu)^2$],

3^{th} moment _t= E [$(x - \mu)^3$], and

4^{th} moment_t= E [$(x - \mu)^4$];

Where, the third moment (called Sharpness in [P2]) is similar to Skewness that is a measure of the symmetry of a distribution. The symmetry for a distribution means that the distribution looks the same to the left and right of the center point. The fourth moment, similar to Kurtosis, is a measure of the "peakedness" of the probability distribution of a real-valued random variable. Higher fourth moment means more of the variance is due to infrequent extreme deviations, as opposed to

frequent modestly sized deviations. In our tests, we found that higher than four moments don't give us new information.

4.5 Time-Frequency Moments Singular Value Decomposition (TFM-SVD)

To overcome the above-mentioned problem of applying the standard SVD to a signal, we introduce a new type of feature extraction method the so-called 'Time-frequency moments singular value decomposition (TFM-SVD)' [P2]. In this new method we suggest the use of eight statistical features of the input signal in time-domain as well as frequency domain. The reason for the use of both time and frequency domains is that if the signal under analysis is a non-stationary signal, as most biological signals and our example BCG are, then we will need a time-frequency analyzing tool to find its most important features, and we can not use the transforms that use either temporal or frequency features alone.

Although there are other feature extraction transforms such as the wavelet transform, we tried to introduce another type of feature extraction method which is able to give results similar to the results of the wavelet transform but having less computational load and is faster. As we explained before, the fast wavelet transform (FWT) is a recursive algorithm and takes time with high computational load to compute wavelet coefficients in higher decomposition level (level six for BCG cycles). Our proposal is to form two matrices: one to store the most elementary statistical parameters of the signal in time domain (Mt) and another one (Mf) to store the most elementary statistical parameters of the frequency domain representation of the signal. After that, we can use singular value decompositions (SVD) to extract singular values (SVs) of those matrixes. We can use these obtained SVs as the most important features of the input signal. The proposed method has the following steps to compute eight features of signal x[n] with length N (samples), which is inside the range [-5,5]:

- Run Fast Fourier Transform (FFT) algorithm to find spectrum (y[m]) of signal x[n];
- Compute four statistical moments of time-series x[n] which are:

  Mean=E[x] = μ ,

  Variance_t=E [$(x-\mu)^2$],

  3th moment _t= E [$(x-\mu)^3$], and

  4th moment_t= E [$(x-\mu)^4$];

- Compute statistical moments of frequency-series y[m]:

  Mean=E[y] = μf ,

  Variance_f=E [$(y-\mu f)^2$],

  3th moment _f= E [$(y-\mu f)^3$], and

  4th moment_f= E [$(y-\mu f)^4$];

- Assemble a new matrix (Mt) with the four following statistical moments of the input signal in time domain (x[n]):

$$Mt=\begin{bmatrix} a * \mu & Variance\_t/b \\ 3^{th}\ moment\_t\ /c & 4^{th}\ moment\_t\ /d \end{bmatrix},$$

  where a, b, c, d are constants with suitable values to scale four elements of Mt to the range [L1,L2]. In [P2] we explained how we can choose a,b,c,d as well as ranges L1 and L2.

- Assemble another matrix (Mf) with the following four statistical moments of the spectrum of the input (y[m]):

$$Mf=\begin{bmatrix} af * \mu f & Variance\_f/bf \\ 3^{th}\ moment\_f\ /cf & 4^{th}\ moment\_f\ /df \end{bmatrix},$$

  where af, bf, cf, df are constants with suitable values to scale four elements of Mf to the range [L3,L4]. In [P2] we explained how we can choose af, bf, cf, df as well as ranges L3 and L4.

- Find singular values (SVs) of the Matrix Mt: SVD (Mt); this will return two SVs for any kind of input signal x[n], because the dimension of Mt matrix is 2x2.

- Find singular values (SVs) of Matrix Mf: SVD (Mf); this will return another two SVs for any kind of spectrum (y[m]) of an input signal, because the dimension of the Mf matrix is 2x2.

It must be mentioned that if we merge Mt and Mf elements in a 1x8 or 8x1 array, it will give us only a SV which is not enough to represent a signal (see definition of SVD).

- Finally, assemble the so-called Time-Frequency Moments (TFM) vector with dimension 4x1:

$$\text{TFM matrix}=\begin{bmatrix} SVD(Mt) \\ SVD(Mf) \end{bmatrix};$$

As can be seen, these steps will return fixed four SVs of time-frequency moments for any type of signal with any size. The properties of this new type of feature extraction method are suitable especially for our application because we would like to optimize and reduce the dimension of the input signals as much as possible.

Singular value decomposition is used because it captures the essential information of a matrix (compression), somewhat similarly to the Eigen values. Referring to our experiences, using SVD after the extraction of the moments gives us better ability to train a neural network for pattern classification.

The choice of TFM normalization parameters depends on the input signal range. In [P2] we presented some formulas to compute normalization parameters.

Chapter 5

Classification methods

5.1 Introduction

A challenging field in machine learning is pattern recognition which goal data (patterns) classification using either a formerly obtained knowledge or essential information/features extracted from the input patterns. The input patterns can be sets of measurements or observations that present essential features in a multidimensional space. A pattern recognition based system has three parts which are: 1) a sensor which senses the electrical (mechanical) activities; 2) a feature extraction step to compute essential features of the input patterns; and 3) a classifier for classifying input data based on the extracted features. The performance of this classifier depends heavily on the availability of input patterns. Neural network based classifiers are based on a learning strategy which can be either supervised or unsupervised. If we give the system a priori labeling of patterns, the learning will be supervised. Otherwise, it creates the classes itself using statistical regularities in the patterns [51].

Classifiers mostly uses statistical (or decision theoretic), syntactic (or structural) methods to classify input data. If the patterns are generated by a probabilistic system, classifiers can use statistical features of patterns (statistical pattern recognition). Structural pattern recognition uses the structural interrelationships of features. Several algorithms can be categorized in this class from a Bayesian classifier to powerful kinds such as neural networks that were used/developed in this study. Bayesian classifiers were not used in this study because the estimation of probabilities for BCG patterns is so complicated [52].

Typical applications of pattern recognition systems are biological signal classification, speech recognition, classification of a text into several categories, recognition of handwritten letters/numbers, face recognition using images of human faces, etc.

5.2 Artificial Neural Networks

Artificial neural networks (ANNs) have become the top biologically inspired methods for solving different computational problems in subjects under the heading Artificial Intelligence. Nowadays, several neural network structures have been implemented from single-layer Pitts-McCullough neurons to complex kinds such as

Kohonen maps, Hebbian learning, Adaptive Resonance Theory (ART) and other modern approaches [52,53].
In another definition, the classification may depend on how signals flow from the input to the output of ANNs. There are two types of ANNs: feedforward structure (signals flow from inputs, forwarding through any hidden units, eventually reaching the output units) and recurrent structure containing connections back from later to earlier layers). The feedforward structure has stable behavior. However, if the network is recurrent it can be unstable, and has very complex dynamics. Recurrent networks are interesting fields for researchers in neural networks, but the feedforward structures have proven to be most useful in solving real problems such as our example BCG. Among different kinds of feedforward neural networks, Multi-layer perceptrons (MLP), and Radial Basis Functions (RBF) are well known and popular [52,53]. So, we used them for testing the abilities and the performance of the developed learning algorithms. However, there are other popular classifiers which could be used for the same purpose such as Competitive Neural Trees (CNeT) [54,55], which explained in section 5.2.3, and Support vector machine (SVM). SVM is a kind of are generalized linear classifier [56,57]. 'They are a set of related supervised learning methods which can be used for regression (a special case of Tikhonov regularization) as well as classification. SVM also called as the maximum margin classifier, because it simultaneously minimizes the empirical classification error and maximizes the geometric margin' [58]. In [59], it is found that SVM with Adaptive Resonance Theory (ART)-MAP can be rather optimal, where ARTMAP itself is another popular method for classification [60]. ARTMAP uses two self-organizing ART neural networks; user defined internal controlling module and internal-ART associative memory. To compare our methods with existing neural network-based classifiers, we chose two classic methods (MLP and RBF) and a popular method (CNeT) which requires a relatively small number of learning cycles. Also, the computational load of CNeT is also lower than of SVM and ARTMAP whilst their classification performance is comparable (see section 6.1- IRIS data classification).

5.2.1 Multi-layer perceptrons

The most commonly used form of Multilayer perceptron (MLP) [52] is a feed forward neural network trained with the back propagation algorithm. It is a supervised neural network and therefore requires a desired response to be trained. It learns how to transform input data into a desired response, and it is widely used for pattern classification. With one or two hidden layers, it can approximate virtually any input-output map. It has been shown to approximate the performance of optimal statistical classifiers in difficult problems. Most neural network applications involve MLP.

5.2.2 Radial Basis Functions (RBF)

RBF-Networks [52] employ neurons that consist of radial basis functions. In contrast to classic multi layer perceptrons the activation of a neuron is not given by the weighted sum of all its inputs but by the computation of a radial basis function. Generally the kernels-Gaussian function is used where u is the input of the neuron, it

is the basis of the neuron, and sigma is the amplitude of the neuron: $\varphi(u;t_k) = \exp(-\frac{1}{\sigma_k^2}\|u - t_k\|^2)$ $k = 1,2,..,M$ where t_k is the center of k[th] kernel and M represents the number of output units. RBF-Networks are feed forward run and consist of one input layer (u), one hidden layer of Gaussian neurons (H), and one output layer (y). The value of an output unit y_i (given a network input u) is computed by $y_i = \sum_{k=1}^{K} w_{ik}\varphi_i(u;t_k) + w_{i0}$ where w_{ik} is the weight of neuron H_i for output y_i and w_{i0} is a general threshold (bias) of output y_i subtracted from the weighted inputs. The Gaussian neurons work as experts for certain areas of the d-dimensional input space. The activation of each neuron depends on its distance to the input vector. Learning algorithms like stochastically learning can be used to adjust the non-fixed parameters of the network including kernel centers, weights and sigma.

5.2.3 Competitive Neural Trees (CNeT)

Competitive Neural Tress (CNeT) was developed by Behnke et al. ([54], [55]) is one of the fast supervised neural networks with high performance (see Fig. 3). A set of similar nodes forms a tree as shown in Fig. 3(a). Fig. 3(b) shows a node in detail. Each node contains 'm' slots and a counter which shows the node age and increase each time an input pattern is presented to the node. The nodes show different behavior when the counter age increases. Each slot stores some items that are: a prototype, counter 'count', and a pointer. The prototypes present clusters of the input patterns. The counter 'count' increases each time the prototype is updated to fit with an input pattern. The pointer points to a childnode assigned to corresponding slot. A slot without any childnode (empty pointer) is called 'terminal slot' or 'leaf'. The internal slots are slots with an assigned child-node.

The growth of the CNeT is based on inheritance for initializing new nodes and can be controlled by forward pruning. It applies unsupervised competitive learning in the node level and clusters the input feature vectors hierarchically. The prototype similar to the input pattern can be found by searching a part of the tree. In [54] and [55] Behnke et al described different kinds of search methods applicable for training as well as for testing (recall).

5.2.4 Problems with neural networks

Supervised neural networks have remarkable performance as practical tools and are effective for a broad range of pattern discrimination and functional approximation applications. As with any type of classifier, the performance of neural networks depends on quality of training material. However, in some applications, the acquisition of a rich training material is not easy and it directly effects on the performance of the classifier [4]. This means that the development of high performance as well as high speed neural classifier which uses a minimum training set with few learning cycles for adjusting weights is difficult but desirable.

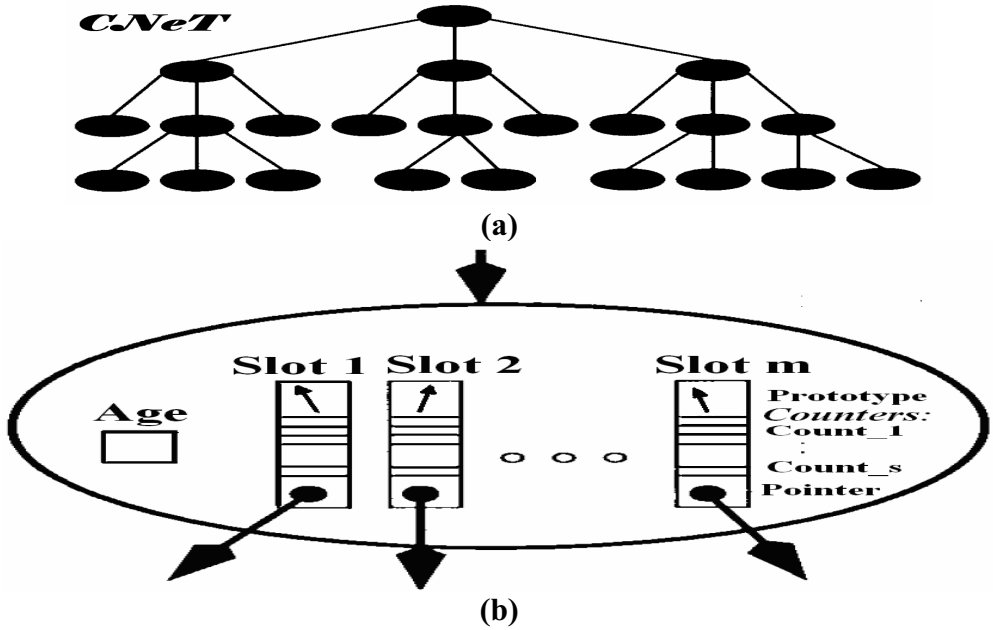


Fig. 5.1. Competitive Neural Tree (CNet) Structure and its nodes: (a) the tree which ellipsoids are its nodes; b) A typical topology of every node in detail.

Most popular neural classifiers like the modular neural networks, wavelet networks, and Neuro-Fuzzy neural networks need to represent heavy training material with a lot of learning cycles. To have a proper neural classifier, we must try to find optimized structure and learning factors for the neural network. However, it is not easy and different structures and parameters must be tried to find the proper values for them and achieve the high performance of pattern classification for the neural network.

Another problem is the fluctuations of classifier performance when different validation input datasets are used. To have minimal variance of performance, sometimes the neural classifier needs to be retrained incrementally whilst new data become available (on-line learning). Most of the well-known supervised neural networks must be retrained with the composite data obtained by combining all the data accumulated thus far. This combination, results in the loss of all previously acquired information, commonly known as catastrophic forgetting. Furthermore, this approach may not even be feasible if the original dataset is no longer available [4].

On the other hand, unfortunately, to achieve an optimized neural classifier, designer will have more labor. Also, learning new information incrementally without forgetting any previously acquired knowledge is not possible, unless all data are memorized. These kinds of problems cause decreased reliability of neural networks. Therefore, we need some improvements in this field so as to create reliable neural classifiers with high performance across different validation tests.

During the past several years, a large number of artificial neural networks either supervised or unsupervised with new structures/ learning algorithms have been developed. Some of them are Adaptive Resonance Theory (ART), neural trees, modified Hopfield, and learner ++ [4]. Most of the existing methods perform remarkably well when optimized learning parameters, rich training material and

enough learning cycles are used.

Methods that do not deal with such important issues may potentially give us misleading information. Another limitation of the existing techniques concerns their degree of success in the case of validation test with new data, not represented yet to net. Other limitations are their ease of hardware and/or software implementation, their stability across different patterns (generalization ability), and their suitability for real-time applications as well as learning incrementally from new data.

5. 3 Combined Learning Network (CombilNet)

To have a reliable neural network with a high classification performance, high learning speed, incremental learning ability and easy to implement characteristics, we introduce a new method so-called Combined Learning Network (CombilNet) in this section. As can be seen in Fig. 5.2, this kind of learning algorithm classifies input samples on two levels. At the first level, the pre-classifier classifies the input samples primarily to M arbitrary classes. During training mode, the post-classifier, which is a special array called Affine Look-up Table (ALT) with M elements, stores the label of the corresponding input sample (y with a value between 1 and N) in the address equal to the index of the pre-classifier winner (j). In the final step, CombilNet classifies the input dataset to N defined classes which should be smaller than M. If N>=M, using another stage would be useless. In the running (testing) mode, the content of an ALT cell with address equal to the index of the pre-classifier winner (j), which was saved during training mode, will be read. The read value (y) declares the final class that input data belongs to. This means that the input sample is classified to a final class with label y.

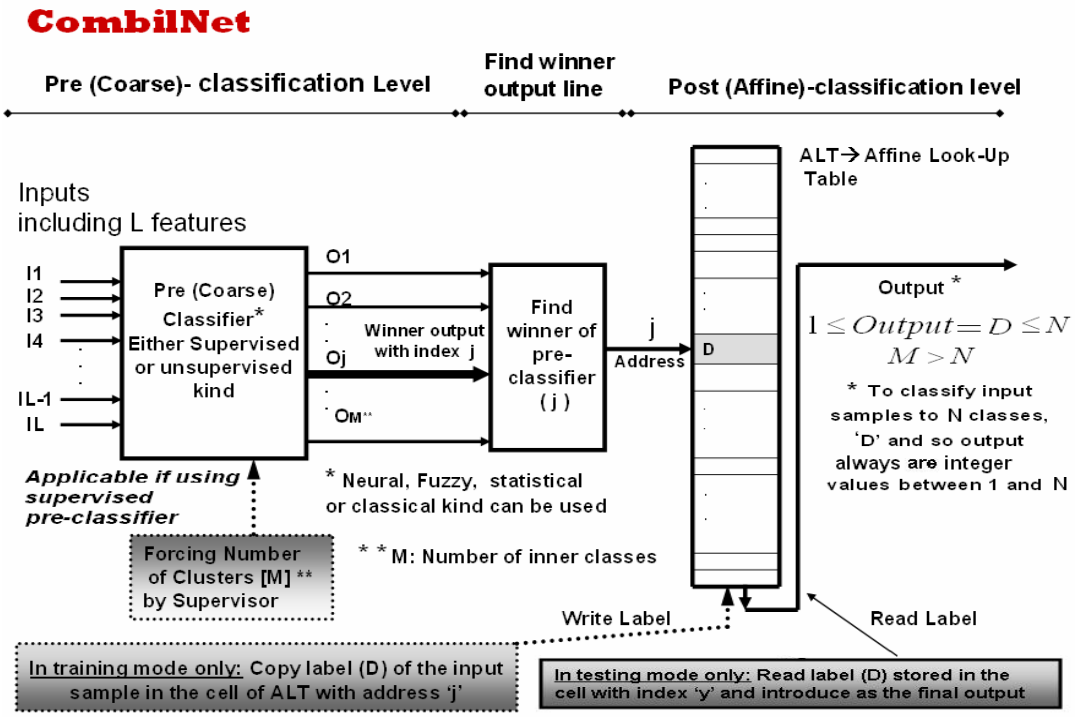


Fig. 5.2. CombilNet structure that is a supervised classifier and classifies input dataset to M inner classes in the first step (pre-classification level) and to final (N) classes (post-classification level).

Pre-classifier

As can be seen in Fig. 5.2, this stage is used to classify the input samples to M inner classes. Output of the pre-classifier is $o_j = F_j(X)$ where F and j are function of the pre-classifier and index of the pre-classifier winner. In general, any unsupervised (supervised) kind of neural network, statistical, or fuzzy classifier can be used as the pre-classifier of the CombilNet. The M must be set in advance if we use the supervised kind. If we use an unsupervised pre-classifier such as Hopfield neural network, this pre-classifier in a self-organized process will set M.

A growable pre-classifier structure of CombilNet can protect previously acquired knowledge during the incremental learning phase. Indeed, the incremental learning ability comes from the cluster growing structure of the pre-classifier. Suppose enough new patterns are presented to the CombilNet. If new patterns don't belong to present inner classes, new inner classes will be generated by the pre-classifier coming from new data. The label of new classes will sit in the not-occupied cells. If new labels don't belong to existing classes, this means that new classes are introduced to the ALT. On the other hand, if new patterns belong to existing inner classes, their labels will sit in some of the previously occupied cells or in some new cells to update old information. This can be happening due to overlap between classes. If we don't use a proper pre-classifier with well-adjusted learning parameters for our application, it will create similar j values for features that belong to different classes. To avoid this problem we have to test different pre-classifiers and find a good one with optimal learning parameters for our specific application.

Moreover, if we use an unsupervised pre-classifier and during the initial clustering it forms a large inner cluster possibly due to lack of dense data in an area, the index of this large inner cluster will be address for an ALT cell (ce1) to store a data label. Now, in incremental learning mode, if new data cover this area more densely, then the unsupervised pre-classifier will create a new inner cluster within the old inner cluster and the index of new inner cluster will be the address for a new ALT cell (ce2) to store input data label. So, the old samples need to be re-organized now into a smaller inner cluster by the pre-classifier. In this case, if the labels stored in the ce1 and ce2 are the same, the input data will be classified to the same final class by the ALT (post-classification level). So, the problem of pre-classifier to classify the input data to two different inner classes won't affect the final classification result. If those labels are not the same, they will be classified to different classes.

It must be mentioned that some neural classifiers with two or more stages have been developed recently, but their structure is completely different. Those kinds of classifiers try to classify input features based on processing and clustering data in two stages. This means that in those kinds of multi-stage classifiers, all the stages use input features (and desired values in supervised learning) to make the final decision. Although the training time in the existing multistage learning algorithms is high, multistage learning algorithms are able to improve the performance of the classification in special applications [61-64].

The performance of CombilNet is dependent on the type of pre-classifier used. Therefore it is not possible to determine the performance of CombilNet in general. An example for CombilNet structure called supervised Fuzzy Adaptive Resonance Theory (SF-ART) is introduced and its performance is evaluated. It uses Fuzzy

adaptive resonance theory (F-ART) as the pre-classifier in CombilNet structure [P4].

It must be mentioned that if we use an unsupervised pre-classifier such as the ART neural network, one problem can occurr. If two samples that have been naturally clustered together into the same cluster whilst they have actually different (known output) labels, the second stage (ALT) can not do any thing because the pre-classifier gives wrong information (same address for both samples). So, the corresponding cell in the ALT will be overwritten. Thus, the final decision for the first presented sample will be wrong.

Affine Look-up Table (ALT)

It is an array with M cells to store NT labels of the corresponding input samples ($\{1 \le D_i \le N\}_{i=1}^{i=NT}$) in the address equal to the index of the pre-classifier winner (j) in the training mode. M, N (<=M) and NT are number of the inner classes, number of the final classes and number of the training samples, respectively. If N>=M, the classifier will not work properly. In the running (testing) mode, a cell with address equal to the index of the pre-classifier winner (j) will be called to pick a label (D_j) that was saved in ALT during training mode. The read value is the final output ($1 \le y = ALT[j] \le N$) declaring the final class that input data belongs to (see Fig. 5.2).

5.3.1 Adaptive Resonance Theory (ART) Networks

The adaptive resonance theory (ART), as one of the fields of neural networks, developed by Carpenter et al., is a popular self-organized classification method [65-67]. Some interesting features of ART and its capabilities have led to its use in different applications in science and technology.

As can be seen in Fig.5.3, any kind of ART-Network can be characterized into three steps: pre-processing, searching (choice and match), and adaptation-levels. The pre-processing level, tries to create an array with a constant number of elements using an input pattern. The format of this fixed size array depends on the kind of ART network we use. When the input pattern is modified to a fixed format in the searching stage, it is compared to the stored templates that are located on the centre of existing clusters. These levels are described in [P4] in more detail.

If the degree of similarity between the current input pattern and the best fitting template (J) is at least as high as vigilance ρ (typically limited to the range [0,1]), this template is chosen to represent the cluster containing the input. The template then adapted by, e.g., slightly shifting the template's values toward the values of the input array. If similarity between input pattern and best fitting template does not fit into the vigilance interval [ρ,1], a new cluster has to be installed where the current input is most commonly used as the first template or cluster centre.

The clustering performance of ART-networks is not well documented in the literature, but Thomas Frank et al. [65] concentrated on the comparative analysis of the clustering properties and the performance of several variants of ART-networks. ART uses single prototypes to internally represent and dynamically adapt clusters. On the other hand, it uses a minimum required similarity between patterns that are clustered within one cluster. The resulting number of clusters then depends on the

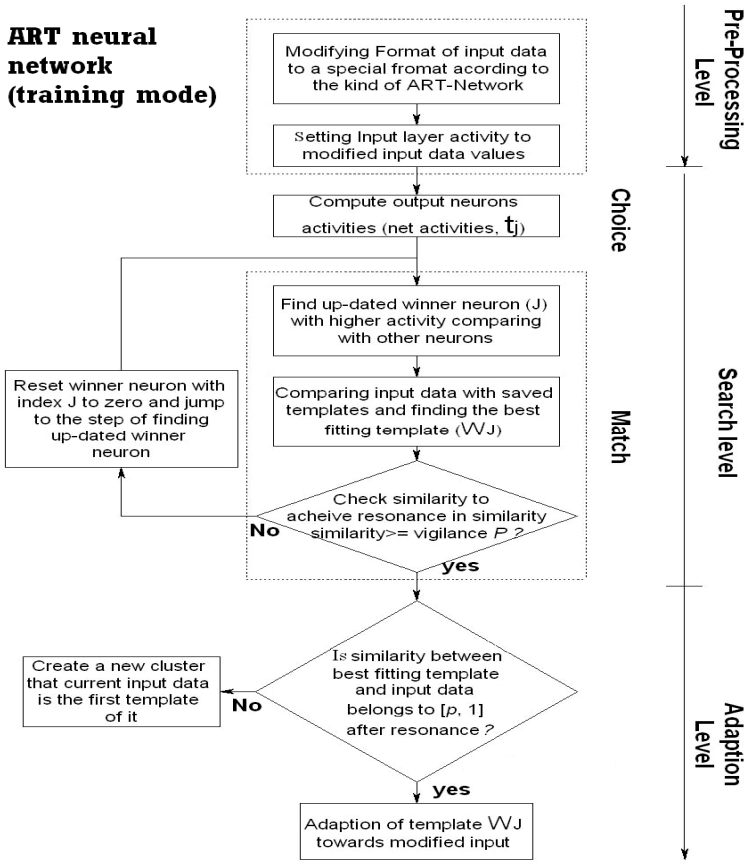


Fig 5.3. Learning Algorithm of ART-Network in three stages [P4].

similarity between all input patterns, presented to the network during the training cycles.

The performance of any kind of classifier depends not only on the network architecture and parameters, but also on the dimensionality and nature of the data to be classified. In this case, all kinds of ART-Networks are very sensitive to the vigilance parameter ρ rather than the nature and the dimension of the input data. This problem causes reduced reliability of this net and decreases its popularity.

5.3.2 Fuzzy Adaptive Resonance Theory (F-ART) Networks

Among different kinds of ART-networks, the Fuzzy ART (F-ART) Network has better performance compared to other kinds in the interpretation of a given dataset. In this kind of ART network, the 'fuzzy AND logic' is used in the second stage to find similarity between input data sample and stored templates. These levels for F-ART are described in [P4] in more detail.

5.3.3 Supervised Fuzzy Adaptive Resonance Theory (SF-ART)

To have a supervised classifier using F-ART, we can use this self-organizing (unsupervised) neural network as the pre-classifier stage to develop a special CombilNet structure called Supervised Fuzzy Adaptive Resonance Theory (SF-ART)

[P4]. Fig. 5.4 shows SF-ART structure. The unsupervised F-ART is too sensitive to the adjustment of similarity parameter (ρ) and learning factor (η). We solved this problem in this new structure (SF-ART). To store corresponding labels of the input patterns in the ALT of SF-ART with a high resolution, we found that a value which is below one but as close to one as possible is the best for vigilance parameter (ρ). To converge SF-ART after a few learning cycles (fast learning mode), the learning factor (η) must be set to 1. With these values SF-ART will be an automatic classifier and free from any adjustment of net parameters. The effect of ρ and η will be discussed further in chapter 6.1.

5.4 QuickLearn Algorithm

To have a learning algorithm with high classification performance across different validation tests and faster than present neural networks with incremental learning ability, we invented a new kind of the supervised learning algorithm so-called QuickLearn [P3]. It has good characteristics based on different structure compared to the existing learning algorithms. QuickLearn is free from any adjustment of the stopping criteria because it needs only one pass of the training set to the net, no need to have second or more learning cycles. Fig. 5.5 shows the structure of the QuickLearn algorithm. It has two levels as explained in the following:

Mapping Level

The first level is the mapping that uses an arbitrary multi input-single output Mapping Function (MF) with fixed weights. This level is used to map the input samples from L-dimensional to one dimensional space. Output of the mapping

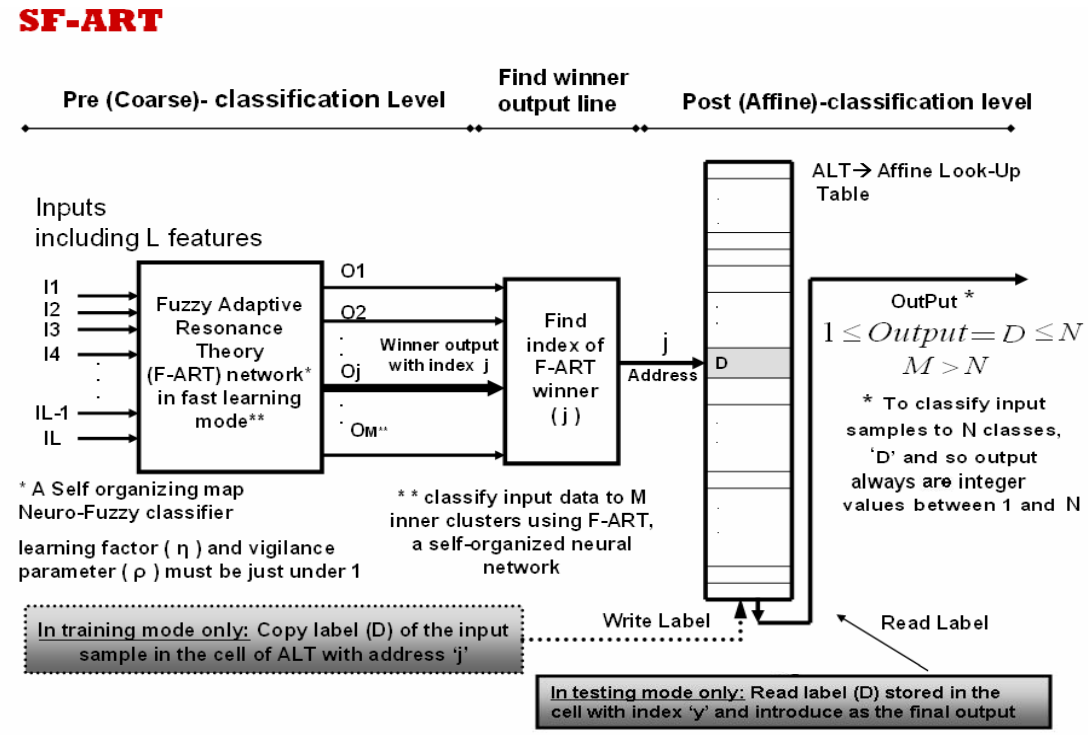


Fig. 5.4. Supervised Fuzzy Adaptive Resonance Theory (SF-ART) structure.

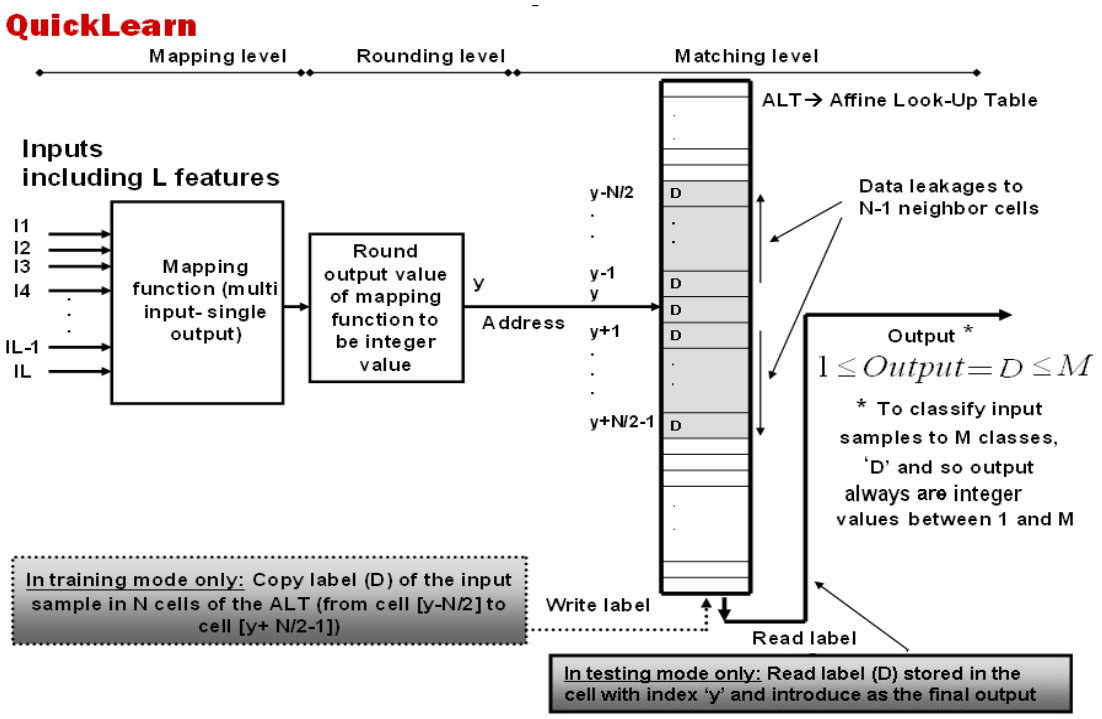


Fig. 5.5. Structure of QuickLearn algorithm.

function is $y = F(x_1, x_2, .., x_L)$. In general, we can select a suitable mathematical function for this map whilst its complexity depends on input data complexity. Indeed, QuickLearn uses only a fixed mapping function without any adaptation of its weights during training phase. So, these kind of fixed maps are only a shifting and then scaling of the input data before representing to the ALT, second stage of QuickLearn.

To map input samples from L dimensional data space to a scalar value, there are several polynomial functions. The simplest ones are linear ($\sum_{i=1}^{i=L} \alpha_i . x_i$) and bilinear ($\sum_{i=1}^{i=L} \alpha_i . x_i^2$) or higher order maps, in which the input samples must be normalized into a range [-1,1], L is the number of inputs for the net, $\{\alpha_i\}_{i=1}^{i=L}$ are weights for the mapping function, and $\{x_i\}_{i=1}^{i=L}$ are the input samples.

To initiate the mapping function weights, we must select proper values for the $\{\alpha_i\}_{i=1}^{i=L}$ constant values. Because the input samples must be normalized to the range [-1,+1], variances on $\{\alpha_i\}_{i=1}^{i=L}$ constant values alone affect the range of the mapping function output. For instance, in this study we found that $\{\alpha_i\}_{i=1}^{i=L} = \{\alpha_{i-1}\}_{i=1}^{i=L} * (10)$ for the linear map and $\{\alpha_i = 1\}_{i=1}^{i=L}$ for the bilinear map give good results. After shifting to positive values the output range for the linear map was in the range [0,T1=1000] and [0,T2=500] for the bilinear map. T must be always equal or larger than the maximum value of map function after

rounding (y). On the other hand, if we choose other values for $\{\alpha_i\}_{i=1}^{i=L}$, map functions' values and their maximum points (so T1 and T2) will change. At the moment the choice of values for α values requires experimentation. A more formal selection process for the α values is a topic for a further study.

Another mapping function is the recently developed CombiMap [P3] which will be explained in Section 5.5. Other kinds of famous polynomial mapping functions are the summation of exponentials, logarithmic, or other kind of mathematical functions which are well-known mapping functions and we can find more about them in [65,66].

Matching Level and Affine Look-up Table

The output value of the map function after rounding will be an integer value inside a range of [0, T]. The value of T is related to values chosen for $\{\alpha_i\}_{i=1}^{i=L}$ factors of linear/bilinear map or permeability factors of CombiMap, which will be explained in the next section. To make a final decision, the final stage, the matching level, uses only an array with T cells called Affine Lookup Table (ALT) to store NT labels of the corresponding input samples ($\{1 \le D_i \le N\}_{i=1}^{i=NT}$) during the training mode. On the other hand, D values are labels for the training material and they depend on the nature of the input data, not ordered by the user. The output value of the map function after rounding is used as a reference address to copy a label for the corresponding input sample (D_y:output desired value) in N cells of ALT (copying the label from the cell ALT[y-N/2] until the cell ALT[y+ N/2-1]) in the training mode (data leakage to N-1 neighbor cells). In the training phase, because the weights of the mapping function (MF) are fixed, QuickLearn doesn't need to repeat the presentation of the training material again. We need only to recall and introduce the value of the cell with index y ($ALT[y] = D_y$) as the final output (final class) in the testing (running) mode, pick a label that was saved in ALT during training mode. The read value is the final output ($1 \le output = ALT[y] \le N$) declaring the final class that input data belongs to ALT cells should be initialized in such a way that all table entries contain the 'default values (zero values)' as output so that if any table cell does not get a value in the training mode, it would contain something instead of random data although our experience shows that almost all cells will be occupied if we present enough input examples and use proper values for T and N. If we present all one features (maximal inputs) to the Mapping function, we can obtain a value that gives estimation for selection of the T. The Value of N is so important. Naturally we don't have yet any rule to choose N and we have to stand in our experience. If we choose a large value for N, the old information will be overwritten. If we choose a small value, some of the ALT cells will be empty by the end of training phase. A proper solution is to test different values for N starting from small values and increase its value to find an optimal value for N.

ALT plays a very important role to protect previously acquired knowledge and to avoid the catastrophic forgetting problem during the incremental learning phase. To be clear, suppose enough new patterns are introduced to the QuickLearn. If these new patterns belong to new classes which are not previously defined to the net, they will sit in the not-occupied cells. But, if they belong to existing classes, they will sit

in some of the previously occupied cells or in some new cells to update old information. This can be happening due to overlap between classes. If we don't use a proper mapping function for our application, it will create similar y values for features that belong to different classes. To avoid this problem we have to test different mapping functions and find an optimal mapping function for our specific application.

However, if new pattern belongs to existing classes, it will manipulate some previously uploaded cells with similar and not new information and it will not affect deeply in classification performance. It must be mentioned that QuickLearn naturally is so sensitive to the type of mapping function and value of leakage size (N) in both learning fresh net and incremental learning mode. So, it is a kind of cooking and we have to test different types of mapping functions such as linear, bilinear, CombiMap and different values for N. My suggestion is choosing N=0.05*T (5% of T size) as an initial value to start searching an optimal value for N.

5.5 CombiMap Transform

To map input features, L dimensional data space, into one dimension (mapping multi-dimensional data to a scalar value) with possibility of keeping some information in a scalar value, a novel map called CombiMap with some good abilities is developed. Mathematical representation of this transform has four terms that are:

$$CombiMap = A + B + C + D.$$

$$A = p_{sum}.\sum_{i=1}^{L} x_i^{ORDER}, \quad B = p_{mx}.Max\ \{x_i\}_{i=1}^{L},$$

$$C = p_{mn}.Min\ \{x_i\}_{i=1}^{L}, \quad D = p_{an}.\sum_{i=1}^{L-1} tg^{-1}(\frac{x_{i+1}}{x_i}).$$

Where ORDER is the rank of this term A. By setting the order of term A to one, it will be a linear measurement of features' placement in the input data space, but by setting it to two, it will measure Euclidean distance between L features of input $X = \{x_i\}_{i=1}^{i=L}$. Input data before presenting to CombiMap must be normalized to the range [-1,1]. Terms A, B, C, and D are unfixed terms controlled by permeability parameters that are p_{sum}, p_{mx}, p_{mn} and p_{an}. Terms B and C are designed to measure the variability of input features in positive direction or negative direction. These terms are suitable to discriminate sharp transient waveforms such as spikes. If most of these spikes are peaked in the positive direction, then term B is recommended, otherwise term C must be used. Term D is designed for input data with almost the same angles that are scattered in a radial direction in data space as well as for input features that are distributed in a complex way, while term B and C are designed for non-complex input data. A level shifting, scaling, and rounding is recommended to obtain positive values in a range [0,T] for CombiMap output. Tuning and initialization of this map is not yet rule based and the permeability parameters must be chosen manually. It is a kind of cooking and depends to our

experiences in the use of CombiMap. Indeed, we have to test different values to find proper values for the applied signal. A more analytical derivation of these values is a topic of a further study.

BCG signal classification using averaged CombiMap (CombiMap$^+$)

It is possible to classify signals such as BCG using CombiMap without any need to train the classifier. For instance, CombiMap can be used for BCG classification after the extraction of features of the BCG. Referring to our experience, the term D must be changed to $D = p_{an} . \cos(\pi . \sum_{i=1}^{L-1} tg (\frac{x_{i+1}}{x_i}))$ if we use the CombiMap for the classification purposes without using any classifier such as neural networks. We called this modified map 'CombiMap+' formerly AliMap [P8]. In the applications in which the input features are scattered in a complex way such as the BCG application, enabling terms B and D, and disabling term C of CombiMap is recommended when the wavelet transform is used as the feature extraction method. For instance, for complex input features such as BCG data this means that term A with second order (ORDER=2) and enabling term B and D have been found to have good properties.

After presenting all segments of the signals of the dataset (BCG cycles of different subjects) to CombiMap, the output will be complex and still not useful. To obtain a quantitative value for patient/subject, the average of CombiMap values for segments (BCG cycles) must be computed, i.e., the computation of averaged CombiMap. Then, the obtained value is shifted and scaled by another map which is

$$Y = \frac{100}{Maximum \quad of \quad CombiMap^+ \quad Averages} * (Averaged \quad CombiMap^+)$$

By this kind of shifting and scaling, we will have quantitative values in a range [0, 100] for the BCG signals of all the patients/subjects. Two thresholds like lines located in y=50, and 100 can be used for the separation between different categories/classes of the BCG signals [P8].

5.6 Classifier performance validation schemes

To evaluate the performance of the classifiers/learning algorithms and check their reliability, there are several well accepted methods such as cross validation and Receiver Operating Characteristic (ROC).

5.6.1 Cross validation and confidence intervals

Cross-validation:

Cross-validation is a way to divide the input data into some subsets. In this way, initial analysis is performed on a single subset and the other subset(s) are employed to validate the initial tests. To check the reliability of the learning algorithms/classifier, we can use the cross-validation test and the partitioning input data into training and testing subsets. There are three popular cross validation tests which are holdout, K-fold and leave-one-out cross-validations [70]. In the first type, training data is chosen randomly from the input data and the rest of the data are employed as the testing data to form the validation data. In the second type, the

original data is divided into K subsets. The second type uses a single input data sample as the (testing) validation data and the remaining input samples as the training data. It must be repeated N times, where N is the number of input samples. So, in this method, each input data sample is used once as the validation data. Finally, the third one which we used in this thesis employs following assumptions (the K-fold cross validation):

- choose N samples randomly for the training set A and M samples to the testing set B;
- repeat the same in the next round: K times taking again N samples for the training set A(i) and M samples to the testing set B(i) but for each i {1..K}. The members of A(i) and B(i) must be chosen differently so that the network is initiated differently each time
- the final result would be the average of the K trials and the confidence interval on the mean but the range of results in each cell of the table (average, range: min...max) can also be reported;

Confidence interval on the mean performance:

In statistics, a confidence interval prepares estimation for a range of values, which is computed from a given set of sample data, to show the uncertainty on an unknown population parameter (performance here). By providing a range of values for an unknown parameter, confidence intervals give us definitely more information than the simple hypothesis tests. The width of the confidence interval means how uncertainty about the unknown parameter. A wide interval means that more data should be gathered before saying anything about certainty of a parameter. If we continue choosing randomly from the same data set (population) N and M samples respectively for training and testing a classifier, then we can compute a confidence interval for each subset using a certain percentage (100(1-α)%) to cover the unknown population parameter, where is (1-α) is called the level of confidence. This confidence interval can be 95%, 90%, 99%, 99.9%, but 95% is a popular confidence level [71].

In [64] a formula is prepared to compute 100(1-α)% Confidence Interval as following: $\overline{X} \pm Z_{\alpha/2} \frac{\sigma}{\sqrt{n}}$ where, $\overline{X}$ and σ respectively are averaged (mean) and standard deviation on classification performance over different testing (validation) datasets, n = number of trials (K in K-fold cross validation test), and $Z_{\alpha/2}$ = Student distribution (Z)-value with an area of $\alpha/2$ to its right (obtained from a table in [72]). This formula can be used when σ is known.

5.6.2 Receiver Operating Characteristic (ROC)

To evaluate the performance of an algorithm like a classifier with two input classes, one way is to find its true positive rate (TP) and false positive rate (FP) for a data set. The rates are the number of target samples, correctly classified as target samples, and non-target samples, incorrectly classified as target over total number of target samples, respectively. However, there are other rates based on TP and FP rates which are true negative rate (TN=1-FP) and false negative rate (FN=1-TP). The TP and TN=1-FP rates are called sensitivity and specificity, respectively. Fig. 5.6 shows

these definitions in a matrix called confusion matrix. For classifying input sample into two classes, each input sample (I) must be labeled to the set $\{P, N\}$ as positive and negative labels. To separate the actual class and predicted class from each other, the labels $\{Yes, No\}$ can be used for the method results [73,74]. This method is so useful in clinical applications to show that a subject is healthy or not (diagnosing purposes), but it can not be used for more than two classes whilst the aim in this thesis was to grouping BCG data into three classes.

To show the tradeoff between TP (sensitivity) and FP (1- specificity) rates of a classifier, a curve called receiver operating characteristic (ROC) developed in the 1950's [73-77] is used. ROC curve is originally from signal detection theory to show with which quality a receiver is able to detect a signal in the presence of noise. ROC curve as an alternative to evaluate the performance of methods might be useful when the comparison of two methods (here, classifiers) on a dataset doesn't give clear results. ROC points can be calculated by varying a threshold value and by computing TP and FP rates for every change. Two typical ROC curves are shown in Fig. 5.7. Similar to the accuracy (=100- error rate) the area under ROC curve (AUC) gives a quantitative measure to evaluate the performance of a method. But, because of the use of different datasets, the comparison of these measures is applicable if the performance of the methods in different uses is similar. In [74] and [77] we can find algorithms and Matlab functions to compute ROC and AUC.

The ROC curve has been used widely in clinical applications to show the effect of changes of a threshold on the numerical results of a diagnostic test [76,78,79]. But, it has been recently used in machine learning and neural network fields for classifying input samples to two classes by changing method parameters/structure, class distributions or misclassification costs [73,74]. In practice to overcome tradeoff between classifier errors e.g. FP and FN we need to find a suitable point on the ROC curve by tuning the classifier. The application of ROC curves for conventional classifiers is better understood than for learning-based methods like neural networks. ROC points can be produced from a neural network by changing net parameter(s). In a MLP neural network, for instance, the bias weight for nodes on the different hidden layers and a threshold value at the output node can be chosen to tune net parameters and generate ROC curve and the area under the curve (AUC), but the most popular one is the threshold value for the output node. Woods et al. in [73] found that using a bias weight for nodes gives a better ROC curve and AUC compared to a threshold value for the output node.

Positive (P)	True Positives (TP)	False Positives (FP)
Negative (N)	False Negatives (FN)	True Negatives (TN)
	Yes	No

Decision:
Hypothesized Class

Fig. 5.6. Common performance measures shown in a matrix called 'confusion matrix' for evaluation results of a method.

To apply ROC curves for three classes, N. Lachiche and Flach, presented a method in 2003 to improve the accuracy and the cost of two-class and multi-class classifiers using two and three dimensional ROC curves [80,81]. Definitely for more than three classes plotting ROC curves is not visually understandable. Because the aim in this thesis is to evaluate the classifiers with three or more classes, ROC curves were not found applicable. As shown in the next chapter only numerical analyses were used based on cross validation tests and confidence intervals.

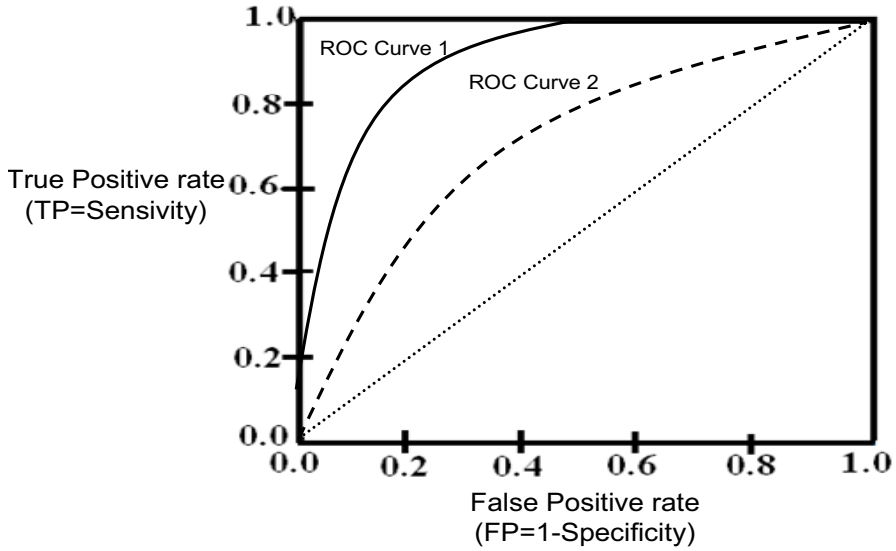


Fig. 5.7. Two typical ROC curves: Roc curve 1 is better than ROC curve 2. A larger area under ROC curve (AUC) means a more powerful classifier.

Chapter 6

Results

6.1 Results for well-known pattern recognition problems

This section evaluates the performance of the developed learning algorithms, SF-ART and QuickLearn as well as existing classifiers on a variety of pattern classification problems. To compare the performance of SF-ART and QuickLearn with existing intelligent classifiers, we used Multi layer perceptron (MLP) [49] and Competitive Neural Tress (CNeT) as one of the fast learning supervised neural networks which learns even complex patterns quite fast and its performance is high [54], [55].

The IRIS Data Set

The SF-ART and QuickLearn algorithms were tested using Anderson's IRIS data set [82,83], which has been used extensively for evaluating the performance of pattern classification algorithms. This data set contains 150 samples of dimension four that are Sepal Width, Sepal Length, Petal Width, and Petal Length. These samples can be divided in three classes (Iris Setosa, Iris Versicolour, and Iris Virginica) representing different IRIS subspecies. Setosa class is far from the other two, which have overlap of their features. Fig. 6.1(a) shows IRIS dataset in terms of two (Sepal Length and Petal Width) dimensions out of four. The 150 samples were randomly split into two sets each containing 75 samples to create training and testing sets. SF-ART and QuickLearn were trained with the training set and its ability was evaluated using the testing set.

As comparisons, Table 6.1 shows the performance of SF-ART, QuickLearn, MLP and CNeT. SF-ART and QuickLearn performed well on the training as well as testing sets. All tests were done in the computer on a 3 GHz Pentium 4 microprocessor. The classification performance was averaged after k-fold (five times) cross validation tests. '$\pm$' shows a 95% confidence interval on the average performance (mean). For MLP we found that two hidden layers with 15 and 10 neurons and 1000 iterations of the training data set had the best performance (97.06 ± 1.06) and one misclassification with the testing set (elapsed time: 2.8 seconds). The SF-ART with η**=1** and vigilance ρ =0.999 (typically limited to the range [0,1]) had a high performance (96.67 ± 1.06) with the testing set and convergence occurred after only six iterations (elapsed time: 0.03 second). The classification performance for QuickLearn was 97.60 ± 1.88 and training took less than one millisecond (so difficult to compute exact elapsed time in C++ or Matlab).

Referring to the table 3.1, Maximum classification performance was 98.71 and 97.73, respectively for MLP (One hidden layer: Nh=5) and SF-ART. It is found that SVM with bagging gives good result (98.66%) [84]. to improve classification performance and stability of a classifier, we can use Bootstrap aggregating (bagging) algorithm [85]. This method is mostly applied to decision tree models, but it can be used with any type of model or classifier. Indeed, class 'Setosa' is linearly separable from other two classes (easy to classify). However, classes 'Virgincia' and 'Versacolour' in some kernel spaces seem to be separable.

Referring to the results obtained for CNet [54-55], after almost 60 adaptation

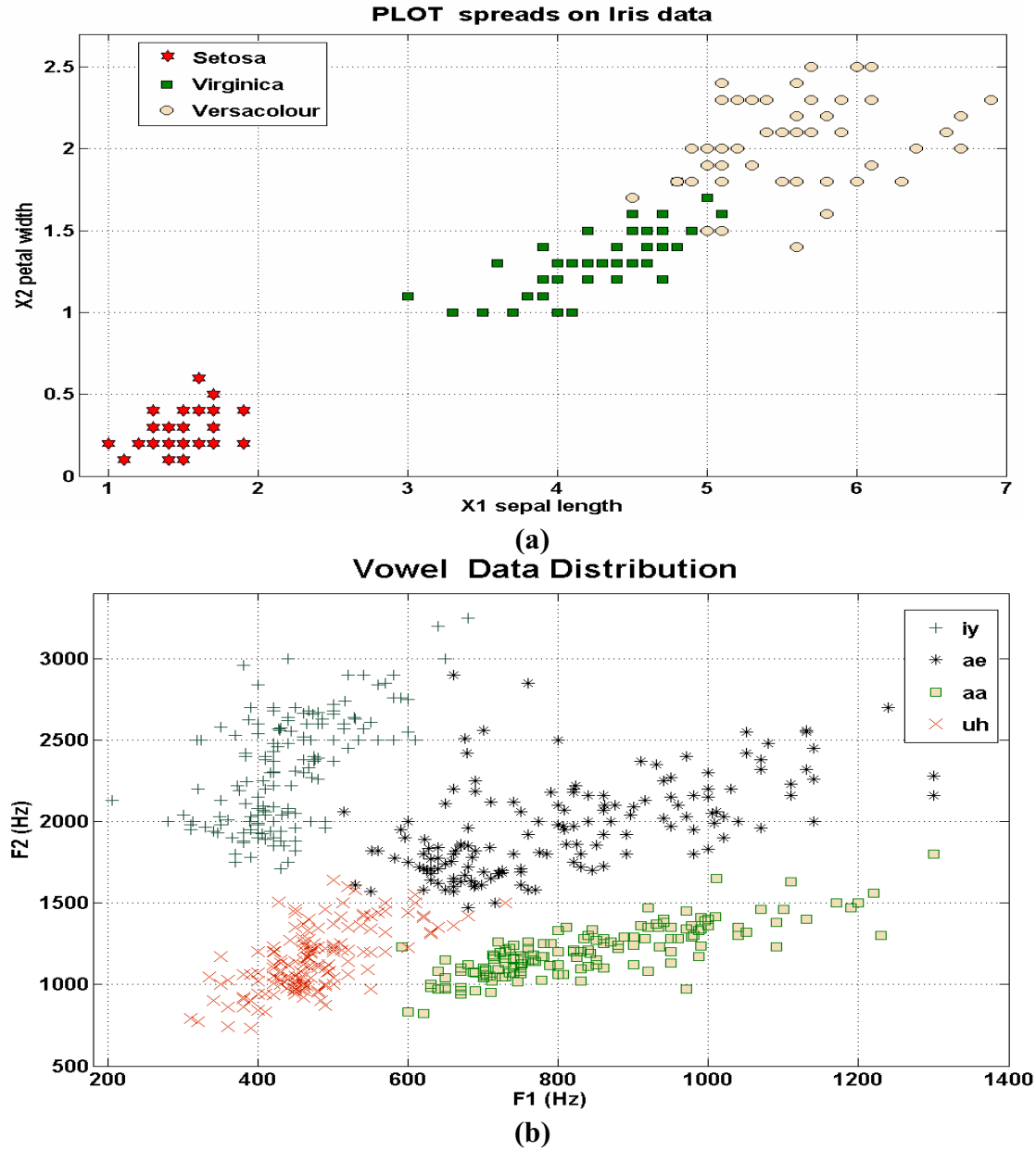


Fig. 6.1. Pattern recognition problems and data sets used for evaluating SF-ART: a) 2-D (petal length- petal width) IRIS Data and its distribution which included 3 classes (Iris Setosa, Iris Versicolour, and Iris Virginica) and represent different IRIS subspecies and b) Vowels Data and its distribution which included 4 classes (iy, ae, aa, uh) and represent different Vowel subspecies.

TABLE 6.1 IRIS data classification to three classes using SF-ART, MLP, CNeT, and QL classifiers. O.P means 'overall performance (averaged)' after k-fold (five times) cross validation tests: '±' shows a 95% confidence interval on the average performance (mean). N.LC means 'number of learning cycles for training'. η = Learning factor, ρ = Vigilance (similarity) parameter. Nh1,2 = Number of neurons for hidden layers 1 and 2. The output ranges were [0,T1=1000] for both the linear and the bilinear maps.

Classifier		Net parameters	% O.P	N.LC
SF-ART		η =1 and $\rho \rightarrow 1$	96.67 ± 1.06	6
CNeT ([54] & [55])		-	94.67	60
MLP with learning rate of 0.001 for all layers		One hidden layer: Nh=5	97.06 ± 1.65	1500
			97.06 ± 1.65	3000
		One hidden layer: Nh=10	96.26 ± 1.06	1500
		Nh1=15, Nh2=10	97.06 ± 1.06	1000
		Nh1=20, Nh2=10	96.53 ± 2.47	1000
QL	linear map with $\{\alpha_i\}_{i=1}^{i=L} = \{\alpha_{i-1}\}_{i=1}^{i=L} * (10)$	N=100, T=1000	96.53 ± 2.47	1
	bilinear map with $\{\alpha_i\}_{i=1}^{i=L} = \{\alpha_{i-1}\}_{i=1}^{i=L} * (10)$	N=200, T=1000	97.60 ± 1.88	1

cycles, the number of incorrect classification (above 4 features) with the training set remained almost constant with some fluctuations. On the other hand, the number of classification errors (above 2 features) with the training set reduced further as the tree kept growing. This is an indication of overtraining, which can be avoided by using the testing-set stopping criterion to terminate the training. Overall, the performance of CNeT in the best situations was above 94.67% on testing set [54-55]. It must be mentioned that IRIS data is designed in such a way that we can not reduce training data size and half of the data set must be used for training and the rest of the data for testing the classifiers.

The Vowel Data Set

Another pattern recognition problem used to check the performance of SF-ART and QuickLearn is a set of 2-D vowel data (608 samples). 'They belong to four classes which are: "IY" as in "eat", "AE" as in "at", "AA" as in "odd", and "UH" as in "two"[86]. Fig. 6.1(b) shows vowel formant data for two repetitions of four vowels by 76 speakers. Formants correspond to resonant frequencies of the vocal tract'. Vowel data set is a popular data to evaluate pattern classification methods whilst there are overlaps between three classes. The 608 samples were split to a training set (300 samples) and a testing set (308 samples).

Table 6.2 shows the comparative analysis and performance of SF-ART, QuickLearn (QL) and Multi layer perceptrons (MLP) applied for the classification of vowel data to four classes. The classification performance was averaged after k-fold (five times) cross validation tests. '±' shows a 95% confidence interval on the average performance (mean). All tests were done on a 3 GHz Pentium 4 computer.

TABLE 6.2 Vowel data classification to four classes using SF-ART, MLP, and QL classifiers. O.P means 'overall performance (averaged)' after k-fold (five times) cross validation tests: '$\pm$' shows a 95% confidence interval on the average performance (mean). N.LC means 'number of learning cycles for training'. η = learning factor, ρ = vigilance (similarity) parameter. Nh1 and Nh2 = Number of neurons for hidden layers 1 and 2. P_{sum} P_{mx}, P_{mn}, P_{ax} are permeability factors of CombiMap. The of CombiMap (CMP) output was in the range [T1=0,T2=T=1000] for all tests.

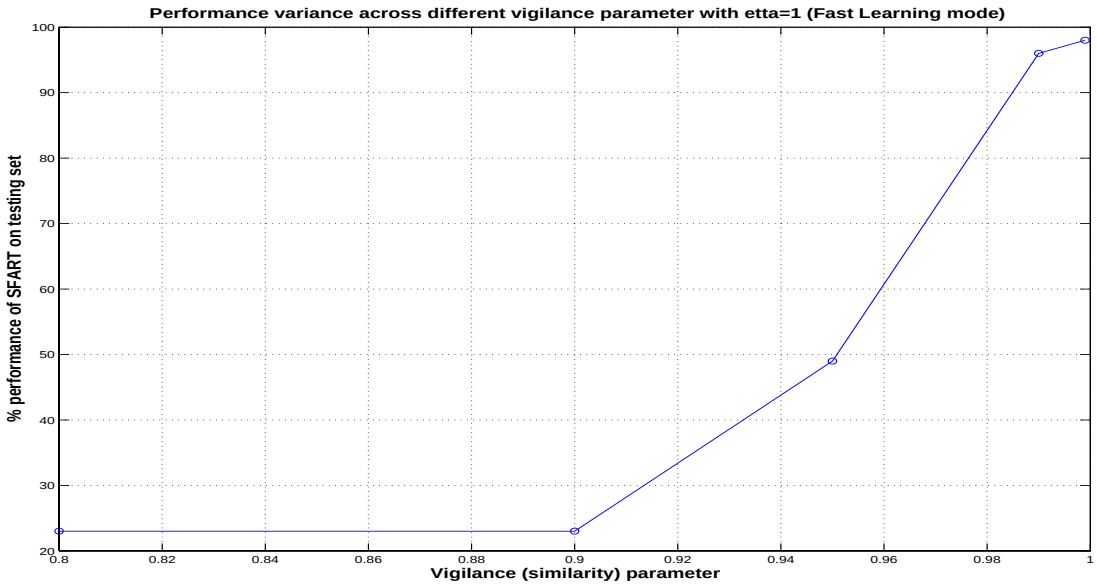
Classifier			Net parameters	% O.P	N.LC
SF-ART			η=1 and $\rho \rightarrow 1$	98.13 ± 0.45	5
MLP with learning rate of 0.001 for all layers			One hidden layer: Nh=5	96.23 ± 1.94	1500
			One hidden layer: Nh=10	91.23 ± 3.29	500
				97.07 ± 1.33	1000
				97.01 ± 1.05	6000
			Nh1=15, Nh2=10	98.37 ± 0.58	2500
			Nh1=20, Nh2=10	98.11 ± 0.55	2500
QL	linear map with $\{\alpha_i\}_{i=1}^{i=L} = \{\alpha_{i-1}\}_{i=1}^{i=L} * (10)$		N=40, T=1000	57.53 ± 3.49	1
	bilinear map with $\{\alpha_i = 1\}_{i=1}^{i=L}$		N=40, T=1000	57.93 ± 3.05	1
	CMP	$P_{sum} = 1,\ P_{mx} = 1$ $P_{mn} = 0,\ P_{ax} = 10$	N=10, T=1000	77.85 ± 2.20	1
			N=20, T=1000	80.06 ± 2.45	1
			N=40, T=1000	80.26 ± 3.77	1
			N=60, T=1000	79.74 ± 3.78	1
			N=100, T=1000	78.44 ± 3.64	1
			N=200, T=1000	64.80 ± 5.59	1
			N=400, T=1000	35.32 ± 12.39	1
		$P_{sum} = 0,\ P_{mx} = 1,$ $P_{mn} = 0,\ P_{ax} = 0$	N=60, T=1000	53.24 ± 5.33	1
		$P_{sum} = 0,\ P_{mx} = 0,$ $P_{mn} = 1,\ P_{ax} = 0$	N=60, T=1000	51.24 ± 2.70	1
		$P_{sum} = 1,\ P_{mx} = 1,$ $P_{mn} = 1,\ P_{ax} = 10$	N=60, T=1000	80.00 ± 4.21	1

Using MLP with two hidden layers the convergence occurred with low error with the training set after above 2500 iterations and it had a performance of 98.37 ± 0.58 with testing dataset with above five misclassifications. Elapsed time for learning was 3.2, 5.4, 26.7 and 34.3 seconds respectively for the MLP with (Nh=5), (Nh=10), (Nh1=15, Nh2=10), and (Nh1=20, Nh2=10) structures.

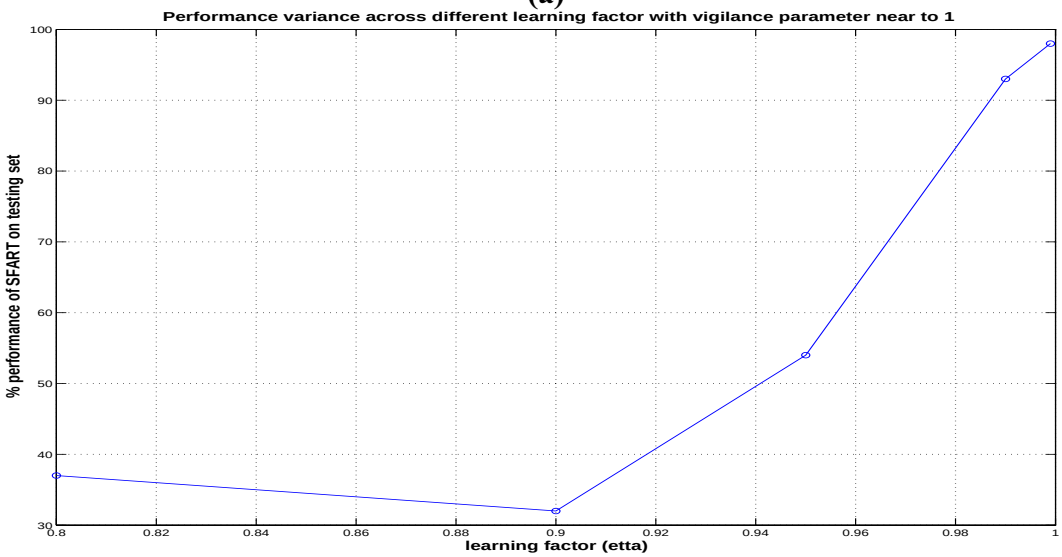
However the SF-ART, with η and ρ closely near to 1, needed only five learning cycles to achieve similar performance (98.13 ± 0.45) with the rest of data (testing dataset). The training time: 250 milliseconds. Fig. 6.2(a) indicates the

performance variances of SF-ART across different values for vigilance parameter (ρ) when the learning factor is constant (η=1). The similar task, in Fig. 6.2(b), is done for different values of learning factor (η) when vigilance parameter (ρ) is constant (near to 1). Fig. 6.3 also indicates the effects of training examples on the performance of SF-ART and MLP. As can be seen, even using a small amount of the training material, SF-ART is more stable than MLP across different volumes of training material. However, in SF-ART, the network capacity for generalization slightly decreased if more than 200 examples (30% of vowels data) were used to train SF-ART (small fluctuation between 97.8% and 98.5% in the classification performance of SF-ART).

In another test, we compared the number of learning cycles and its effects on the



(a)



(b)

Fig. 6.2. SFART Performance variances across a) different vigilance (ρ: similarity) parameter with constant learning factor (η) near to 1 and b) different learning factor (η) with vigilance (ρ) near to 1. The examples for training and testing classifiers are selected randomly to learn patterns efficiently.

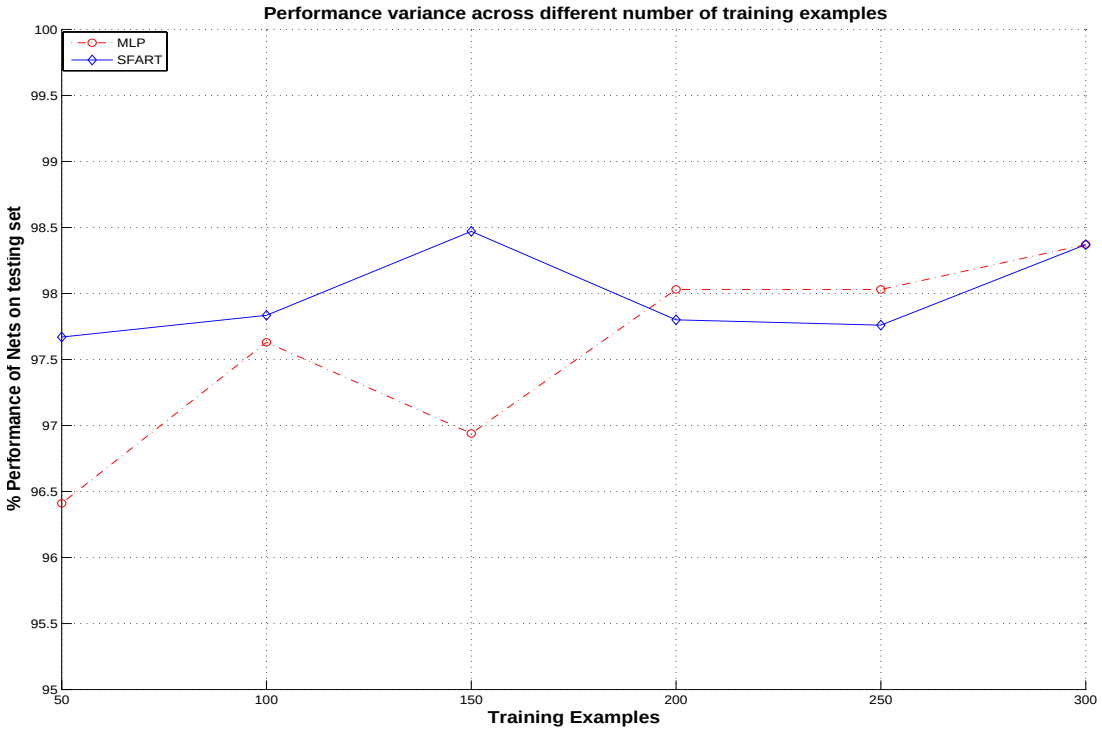


Fig. 6.3. MLP and SFART Performance variances across a different number of training examples. For SFART learning factor (η) and vigilance (ρ: similarity) were chosen near to 1. Also, for MLP with two hidden layers, numbers of neurons for hidden layers were: Nh1=15, Nh2=10. The examples for training and testing classifiers are selected randomly to learn the patterns correctly. The training and testing data sets were the same for MLP and SF-ART.

performance of existing supervised neural networks such as MLP (see Fig. 6.4). In most cases, neural networks need the adjustment of a stopping criterion in the training mode using mean square error, a manual limit or other kinds of criteria. However, SF-ART is free from any adjustment of the stopping criterion because it stops automatically after a resonance occurrence in the training mode. The convergence will be done very fast (after five learning cycles for the vowel data set) because SF-ART uses the fast learning mode for F-ART. Therefore, in Fig. 6.4, there is only one point with position (iteration number=5, performance=98.38%) in the figure for SF-ART. This ability helps SF-ART to be a fully automatic neural network based classifier.

We also tested QuickLearn (QL) net using different types of CombiMap structures, linear, and bilinear maps. As can be seen in Table 6.2, the QL has a somewhat lower classification performance than MLP and SF-ART but its clear advantage is its ability to learn through a single pass of the training data while SF-ART needed five learning cycles. A fresh QL needs a few microseconds to learn. The QL is almost insensitive to the value of N if it is larger than 1. If we use the CombiMap with (P_{sum}=1, P_{mx}=1, P_{mn}=0, P_{ax}=10) and QL with (N=40, and T=1000), the performance to classify vowel data to four classes will be highest (80.26 ± 3.77). Referring to our tests, by choosing $\{\alpha_i\}_{i=1}^{i=L} = \{\alpha_{i-1}\}_{i=1}^{i=L} * (10)$ for the linear map

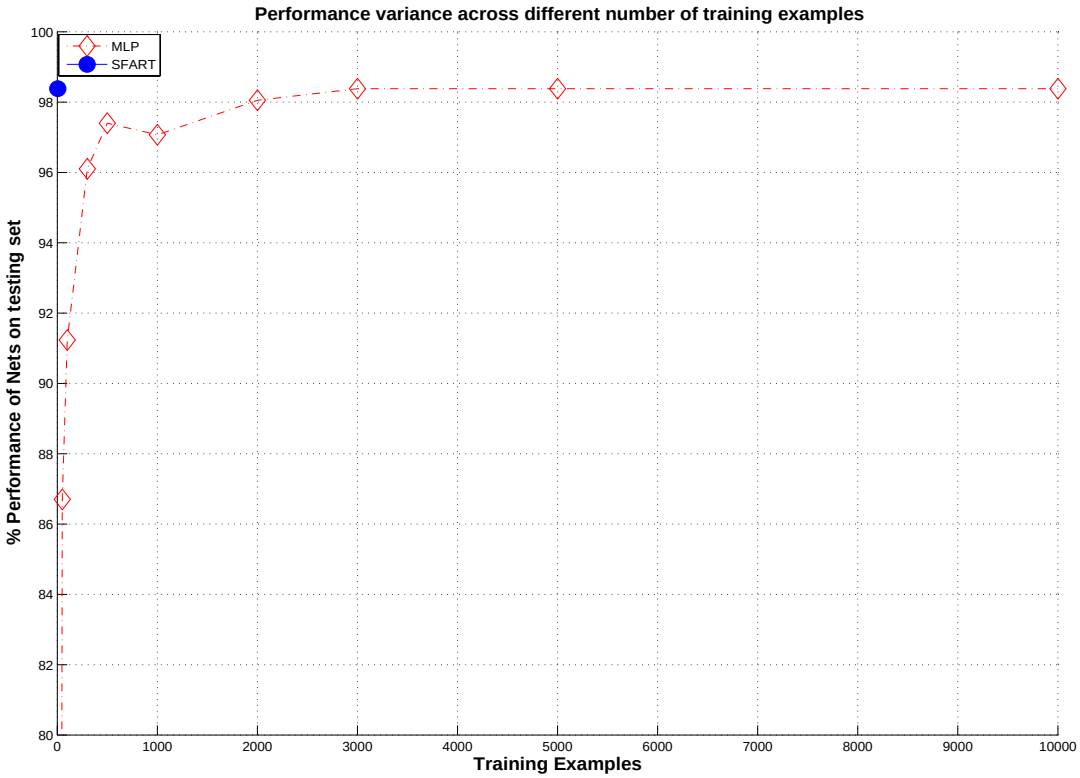


Fig. 6.4. MLP and SF-ART performance variances across a different number of adaptation (iteration) cycles during training mode. For SF-ART learning factor (η) and vigilance (ρ: similarity) were chosen near to 1. Because SF-ART stops automatically (after five adaptation cycles), we have only one point in the figure for SF-ART. Also, for MLP with two hidden layers, the numbers of neurons for hidden layers were: Nh1=15, Nh2=10. The examples for training and testing classifiers were selected randomly to learn the patterns efficiently.

($\sum_{i=1}^{L} \alpha_i . x_i^2$) and $\{\alpha_i\}_{i=1}^{i=L} = 1$ for the bilinear map ($\sum_{i=1}^{L} \alpha_i . x_i$), we get the best results. QL with CombiMap gives better result than QL with linear/bilinear maps.

Behnke et al. [54-55] used IRIS and their own 2-D vowel data with ten classes, instead of the four classes that we used, to evaluet CNeT performance. The ten vowel classes which they used came from: "head", "hid", "hod", "had", "hawed", "heard", "heed", "hud", "who'd", and "hood". The available 608 feature vectors were divided into a training set, containing 300 vectors, and a testing set, containing 308. They compared CNeT performance with other existing methods such as K-Nearest Neighbor (K-NN) classifier and MLP with five and ten hidden units trained using gradient descent. The classification performance for CNeT, K-NN, MLP with five hidden units and ten hidden units, were 82.04%, 75.45%, 76.58% and 80.18% for the testing set, respectively (Table 6.3). As can be seen the CNeT had better performance than MLP and K-NN to classify vowel data to ten classes and learnt faster. But it needed more than 60 learning cycles to converge on the training set with some fluctuations in the number of incorrect classifications. CNeT is a heavy algorithm and the time required for training is high compared to SF-ART and specially QuickLearn.

TABLE 6.3 Ten classes' vowel data classification using CNeT, MLP, and K-NN classifiers [54-55]. O.P means 'overall performance'.

Classifier	Net parameters	% O.P
CNeT	-	82.04
k-NN	-	75.45
MLP	5 hidden units	76.58
	10 hidden units	80.18

Computational load of SF-ART

The computational load of SF-ART is N*Ni*M*L, where N is the number of learning cycles (typically less than 10 cycles for most pattern recognition problems: training time above a few seconds to train a fresh classifier), Ni is the number of training samples, M=7 is the number of mathematical calculations (sum, subtraction, division and multiplication) and L=2 is the number of logical/comparative calculations (>,<,=, Min and Max). This means that F-ART and so SF-ART have a low computational load and the post-classification level is a simple array of memory cells and they are easy to implement on chip [84,85].

Computational load of QuickLearn

The computational load of the QuickLearn algorithm is only M*IL*Ni (M=2) for a linear map and M*IL*Ni (M=3) for a bilinear map. Thus it is easy to implement on chips even on mobile devices. IL is the number of input features (input lines). Notations M and Ni is explained above.

6.2 Results of Ballistocardiogram Classification

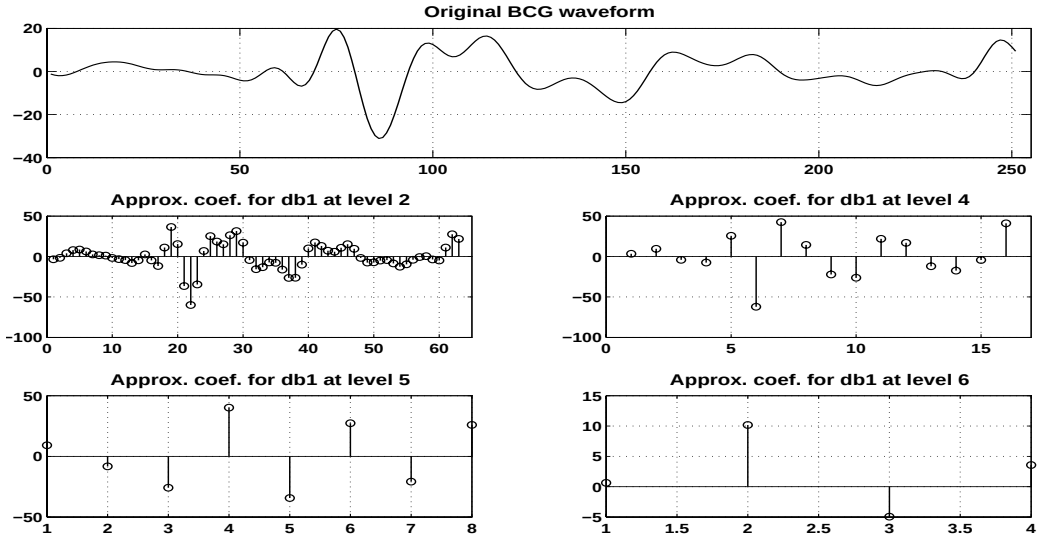
In this section we analyze and compare the classification performance of the methods published in [P1, P2] and [P5-P8] using more validation tests and 95% confidence intervals on mean classification performance. In this study, our aim is only the evaluation of the developed methods from a methodological point of view using BCG signals of a maximum thirty subjects/patients. To design a heart diseases diagnosing system that is reliable for clinical applications, we have to test the developed methods with at least one hundred subjects.

6.2.1 Evaluation with the BCG data set of six subjects

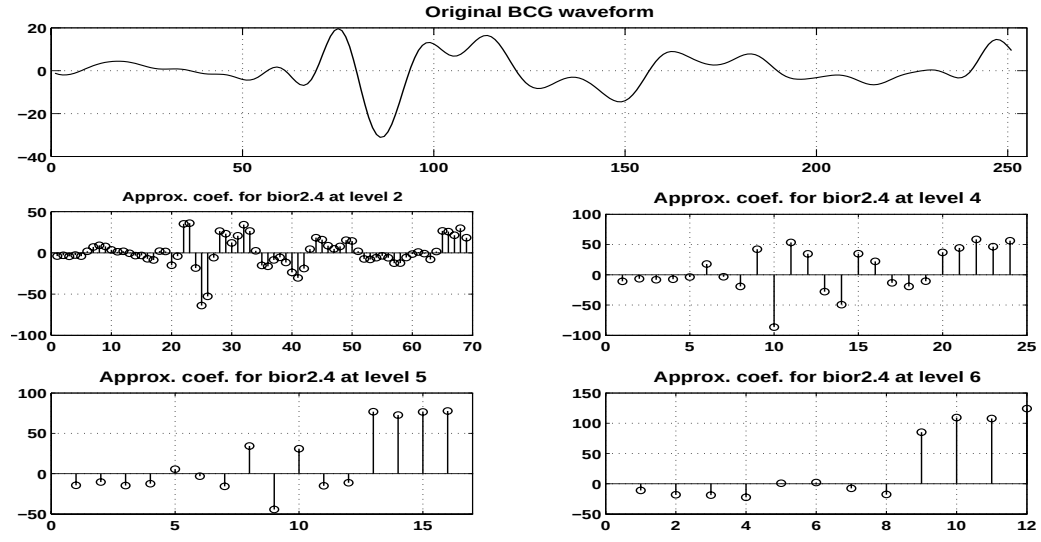
A) Using wavelet transform for BCG feature extraction:

In this study we used both Daubechies and biorthogonal spline wavelets for BCG feature extraction. Our practical experiences showed that using 'P=2' for Daubechies and 'p=2, $\tilde{p}$=4' for biorthogonal spline wavelets are enough and the most important features of BCG waveforms were saved at the iteration level 6 of the FWT, as the maximum level of decreasing signal dimension from 250 samples (duration of BCG cycles) to 4 samples. Parameters p and $\tilde{p}$ are explained in [P1].

Indeed, all WT coefficients from the decomposition stage six were presented as four important features of every BCG cycle of six subjects to the classifier. Fig 6.5 shows wavelet coefficients at four different levels for a typical BCG waveform using both daubechies and biorthogonal spline wavelets. Fig. 6.6 shows a 3-D representation of the first three wavelet coefficients (in level 6), extracted from every BCG cycle, for six typical subjects of three categories: young healthy students aged 20-30 years old (2 subjects), old healthy men aged 50-70 years old (2 subjects) and two old subjects (50-70 years old) with a history of a heart infarct. For every subject of the three categories, the wavelet (daubechies) decomposition stage six gave four features of every BCG cycle. We ignored first eight coefficients at decomposition stage 6 for Biorthogonal wavelets, which are small, to have four samples, similar to daubechies wavelet transform. To use these data as both training and testing data for classifiers, they were normalized, mapped to range [-1, 1], and finally saved randomly into a unique data matrix.



(a)

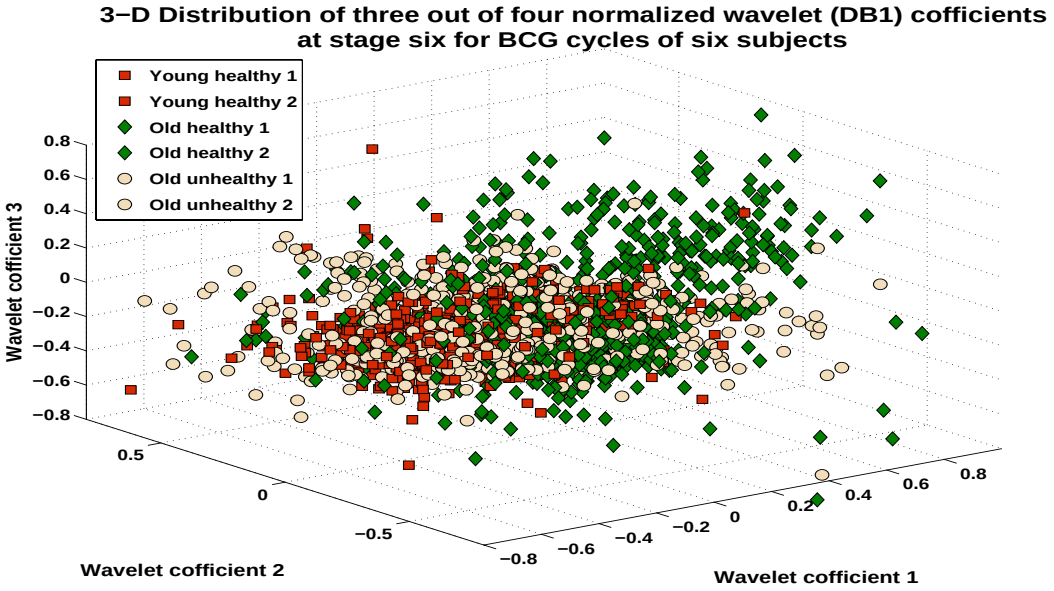


(b)

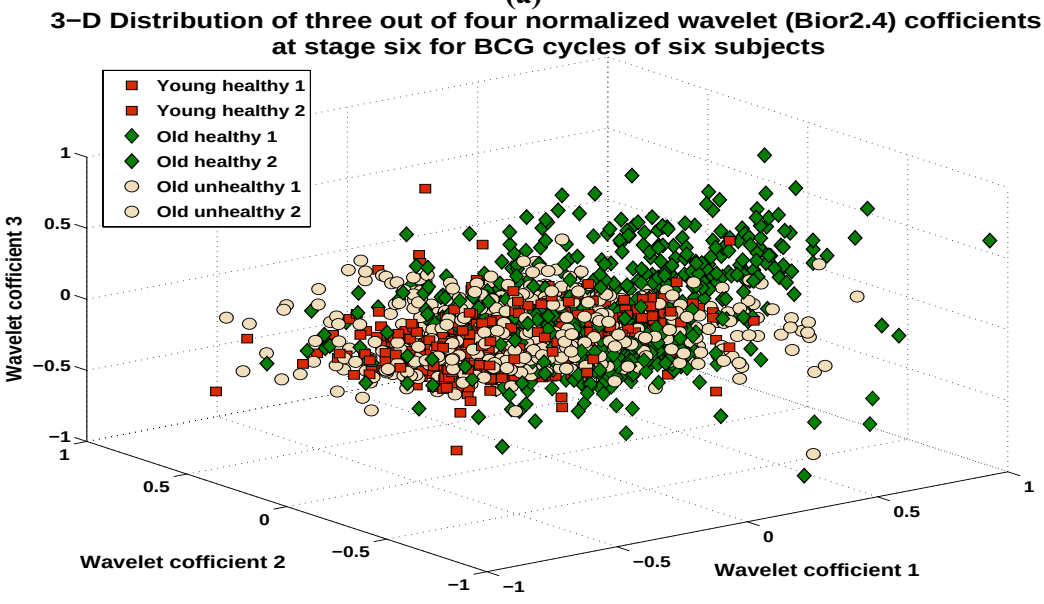
Fig. 6.5 Wavelet coefficients at 4 different levels for a typical wavelet waveform using a) Daubechies and b) Biorthogonal compactly supported.

We applied SF-ART, QuickLearn and Multi Layer Perceptrons (MLP) with structure (4 inputs/3outputs) for the BCG classification of six subjects to the above-mentioned three categories. For MLP, our previous works showed that structure with Tanh(.) to simulate the non-linearity of neurons and two hidden layers (15, and 10 neurons, respectively) is also an optimal structure for the BCG classification [P1, P2], [P5-P8], [89-93].

In this study, the CombiMap is used in the first stage of the QuickLearn for the classification of the BCG records. According to our experience, enabling term B and



(a)



(b)

Fig. 6.6. 3-D Representation of three out of four wavelet coefficients (level 6) for BCG cycles of typical six subjects of 3 categories: young normal, old normal, and old abnormal subjects using a) Daubechies which gave four coefficients at decomposition stage six and b) Biorthogonal wavelets which gave twelve coefficients at decomposition stage six (first eight coefficients was ignored to have only four samples similar to Daubechies).

D, and disabling term C is recommended for applications in which the input features are scattered in a complex way such as our application. We used term A with second order (ORDER=2) and $P_{sum}=1,\ P_{mx}=1,\ P_{mn}=0,\ P_{ax}=0$. To obtain positive values in a range [0,T=1000] for CombiMap values, a level shifting, scaling, and rounding is recommended before presenting to the ALT with T cells in QL. Referring to our tests, if we use data leakage length N=10, we will get the best classification performance for BCG patterns [P6].

We used a small part of data (300 BCG cycles) for training the SF-ART, the QuickLearn, RBF and the MLP nets. All tests were done in a 3 GHz Pentium 4 computer. Learning speed is low for MLP with back propagation and convergence occurred after 10000 learning cycles. Training time for learning was around five minutes while SF-ART needed 470 milliseconds (seven learning cycles) for convergence. For the RBF with Nhidden=20 and Learning factors (0.01, 0.01, and 0.001 respectively to adjust kernels centers, weights and sigma) convergence occurred after 1000 iterations (elapsed time: around 5.00 seconds). The QuickLearn needed only a few microseconds for training. This is because QuickLearn is free from repeating representation of training set to the net and one cycle is enough. But its performance is low (77% for BCG classification).

In testing mode, the rest of the data (2000 BCG cycles) were used to check the performance of the neural networks, not using the same data for training or testing the system although the BCG cycles were obtained from the same subjects as the training data. Table 6.4 shows the performance of the classifiers under test. It can be seen that their performance is high, but SF-ART and QuickLearn learnt very fast. When the highest proportion of cycles in a class is taken as the classification result of the subject, no classification errors are made for both MLP and SF-ART. However, the QuickLearn classified one subject of class 3 to class 2 and one subject of class 2 to class 3. It had a performance of 77.30 ± 1.71 on testing dataset. If we train the QuickLearn partially using the BCG cycles of individual subjects, the classification performance for the rest of the subjects will be low (local learning). This problem was also reported for BCG classification of the same six subjects using Radial Basis Functions (RBF) [P2], but QuickLearn has a better local classification performance than the RBF that gave 100% classification errors for the subject 5. The RBF had a performance of 66.72 ± 7.03 with the testing set that is lower than the performance of QuickLearn. It must be mentioned that the MLP is able to learn both local and general information with high performance while the RBF and the QuickLearn cannot (see Table 6.4).

B) Using TFM-SVD for BCG feature extraction:

In this study we also used the TFM-SVD for the BCG feature extraction of six subjects under test [P2], [92] before presenting to the neural networks. The TFM-SVD was applied on the signal cycle and its FFT without any wavelet decomposition. Indeed, WT and TFM-SVD were used separately for feature extraction. For every subject, the TFM-SVD gave us four features of every BCG cycle. These data were normalized, mapped to the range [0, 1], and finally saved randomly into a unique data matrix. We used a small part of the data for the training of neural networks (300 BCG cycles) and the rest of the data (2000 BCG cycles) for testing the performance of the classifier. In this study there were no excluded

subjects for testing and we used the same subjects for both training and testing the MLP, SF-ART, QuickLearn and RBF classifiers.

As can be seen in Table 6.5, the MLP and SF-ART performed clearly better than the RBF and QL networks. If the MLP and SF-ART had made the decision to which class the subject belongs, it would have classified every subject correctly. This

TABLE 6.4 Classification of the BCG data of six subjects to three classes using SF-ART (SF), MLP, RBF, and QuickLearn (QL) classifiers and Biorthogonl wavelets. C, and SBJ mean 'Class number' and 'subject number', respectively. O.P means 'overall performance (averaged)' after k-fold (five times) cross validation tests: '$\pm$' shows a 95% confidence interval on the average performance (mean). Structures: SF, η and $\rho \rightarrow 1$; MLP, Nh1=15, Nh2=10, learning rate of 0.001 for all layers; RBF, Nh=20, learning factors to adjust kernels centers, weights and sigma were 0.01, 0.01, and 0.001 respectively; QL, N=10, T=1000 and $P_{sum} = 1,\ P_{mx} = 1,\ P_{mn} = 0,\ P_{ax} = 0$.

Classifier			C1	C2	C 3	O.P
SF	C1	SBJ1	97.93 ± 2.28	2.07 ± 2.28	-	
		SBJ2	94.94 ± 2.83	5.06 ± 2.83	-	
	C2	SBJ1	-	61.72 ± 36.8	38.28 ± 36.8	
		SBJ2	-	89.87 ± 1.53	10.13 ± 1.53	
	C3	SBJ1	-	21.34 ± 6.90	78.66 ± 6.90	
		SBJ2	-	3.46 ± 0.94	96.54 ± 0.94	
	O.P%					94.70 ± 0.86
MLP	C1	SBJ1	96.26 ± 3.06	3.74 ± 3.06	-	
		SBJ2	83.13 ± 5.99	16.87 ± 5.99	-	
	C2	SBJ1	-	99.40 ± 0.26	0.60 ± 0.26	
		SBJ2	-	77.80 ± 4.29	22.20 ± 4.29	
	C3	SBJ1	-	3.54 ± 1.37	96.46 ± 1.37	
		SBJ2	-	5.20 ± 0.48	94.80 ± 0.48	
	O.P%					90.65 ± 1.51
RBF	C1	SBJ1	75.13 ± 33.5	24.18 ± 33.5	-	
		SBJ2	76.73 ± 40.9	23.27 ± 40.9	-	
	C2	SBJ1	-	95.13 ± 9.84	4.87 ± 9.84	
		SBJ2	-	69.53 ± 27.6	30.47 ± 27.6	
	C3	SBJ1	-	100.0 ± 0.0	0.0 ± 0.0	
		SBJ2	-	61.2 ± 3.72	38.8 ± 3.72	
	O.P					66.72 ± 7.03
QL	C1	SBJ1	64.06 ± 7.13	35.94 ± 7.13	-	
		SBJ2	86.34 ± 6.96	13.66 ± 6.96	-	
	C2	SBJ1	-	60.40 ± 41.0	39.60 ± 41.0	
		SBJ2	-	35.00 ± 6.34	65.00 ± 6.34	
	C3	SBJ1	-	86.44 ± 9.35	13.56 ± 9.35	
		SBJ2	-	32.60 ± 5.17	67.40 ± 5.17	
	O.P%					77.30 ± 1.71

means that more than 50% of the BCG cycles for every subject were always in the right class when BCG cycles were selected randomly. Thus we can say that the entire recordings were classified correctly using the MLP and SF-ART networks. This means that the local minima or non-linear disturbances have only a small effect on the overall performance of MLP and SF-ART. In terms of our investigations, this performance did not increase if more epochs or adaptation cycles (applicable only for MLP) were used during the training phase for the MLP and SF-ART networks.

The RBF would have misclassified one subject and would have nearly misclassified another. Again, the results did not improve if more epochs or adaptation cycles were used during the training phase for the RBF network. The QuickLearn with TFM-SVD had a lower performance (51.30 ± 2.42) than the RBF with TFM-SVD (64.20 ± 4.83). Both RBF and QuickLearn had a low local performance with both TFM-SVD and wavelets feature extraction methods.

If we compare the results of using the TFM-SVD method for BCG feature extraction with our results using the wavelet transforms (WT) to extract the essential features of the BCG cycles [P1], [P3-P8], we see that the results for both of these methods are comparable. The overall performance of the MLP and SF-ART networks using WT were above 90.65 ± 1.51 and 94.70 ± 0.86, whilst these are 84.74 ± 0.82 and 79.40 ± 1.06 using TFM-SVD for the same (six) subjects under test. Although the performance using the TFM-SVD was more than 5% lower than using the WT and MLP, the TFM-SVD is faster and easier to implement than the WT. This is because the WT is a multi-resolution time-frequency transform and we must use an iterative algorithm, the so-called Mallat Algorithm, to find a higher resolution in the frequency domain and low resolution in the time domain [44].

6.2.2 Evaluation with the BCG data set of thirty subjects

We also checked BCG classification performance for MLP and SF-ART using 30 subjects [P1]. Referring to the above-mentioned results for six subjects, these two classifiers were the best candidates and we excluded the QL and RBF from this test. To train and then test the neural network, we divided the 30 subjects into two sets by randomly selecting an equal number of subjects from each subject group: a set of 21 subjects (seven from each group) for training and testing, and nine (three from each group) for validation test. From each subject in the 21 subject set we used 70% of data (6000 BCG cycles) for training the ANN and the rest of the data (2500 BCG cycles) for testing the performance of the ANN classifier, not using the same data of 21 subjects for training or testing the system. Then, to validate the results, we applied our method for the set of nine new subjects that were not presented in the training phase to the net. To perform a K-fold cross validation, we repeated the previous procedure (random division of subjects into two sets, training, testing and validation) five times, so that the network was initiated differently each time.

The optimization of the size of the MLP network was done for all the five trials using different net structures. To do these tests we used a learning rate of 0.001 for all layers because it gave us better results for all MLP structures than the other values we tested. The final result is the average of the five trials as can be seen in Table 6.6 for both SF-ART and different MLP structures. We found that the size with two hidden layers (structure 2: 15, and 10 neurons, respectively), is rather optimal. In order to make sure that the found optimum was a global one, also a third hidden

layer structure was tested and the number of neurons optimized. The third hidden layer did not improve the results anymore and therefore the two hidden layer structure was chosen as the final one. The increase of the number of hidden layers also increases system complexity, computational load, and training time.

TABLE 6.5 Classification of the BCG data of six subjects to three classes using SF-ART (SF), MLP, RBF, and QuickLearn (QL) classifiers and TFM-SVD. C, and SBJ mean 'Class number' and 'subject number', respectively. O.P means 'overall performance (averaged)' after k-fold (five times) cross validation tests: '±' shows a 95% confidence interval on the average performance (mean). Structures: SF, η and $\rho \rightarrow 1$; MLP, Nh1=15, Nh2=10, learning rate of 0.001 for all layers; RBF, Nh=20, learning factors to adjust kernels centers, weights and sigma were 0.01, 0.01, and 0.001 respectively; QL, N=10, T=1000 and $P_{sum} = 1,\ P_{mx} = 1,\ P_{mn} = 0,\ P_{ax} = 0$.

Classifier			C1	C2	C 3	O.P
SF	C1	SBJ1	89.20 ± 3.75	10.8 ± 3.75	-	
		SBJ2	85.13 ± 2.12	14.87 ± 2.12	-	
	C2	SBJ1	-	86.53 ± 1.61	13.47 ± 1.61	
		SBJ2	-	71.73 ± 4.18	28.27 ± 4.18	
	C3	SBJ1	-	41.74 ± 6.82	58.26 ± 6.82	
		SBJ2	-	19.60 ± 2.41	80.40 ± 2.41	
	O.P%					79.40 ± 1.06
MLP	C1	SBJ1	96.06 ± 1.93	3.94 ± 1.93	-	
		SBJ2	92.73 ± 2.30	7.27 ± 2.30	-	
	C2	SBJ1	-	92.33 ± 1.81	7.67 ± 1.81	
		SBJ2	-	76.20 ± 8.12	23.80 ± 8.12	
	C3	SBJ1	-	47.94 ± 18.7	52.06 ± 18.7	
		SBJ2	-	13.80 ± 1.82	86.20 ± 1.82	
	O.P%					84.74 ± 0.82
RBF	C1	SBJ1	92.13 ± 7.90	7.87 ± 7.90	-	
		SBJ2	96.86 ± 4.55	3.14 ± 4.55	-	
	C2	SBJ1	-	66.33 ± 7.31	33.67 ± 7.31	
		SBJ2	-	90.26 ± 2.38	9.74 ± 2.38	
	C3	SBJ1	-	100.0 ± 0.0	0.0 ± 0.0	
		SBJ2	-	87.5 ± 14.28	12.5 ± 14.28	
	O.P					64.20 ± 4.83
QL	C1	SBJ1	40.0 ± 13.70	60.0 ± 13.70	-	
		SBJ2	81.34 ± 4.18	18.66 ± 4.18	-	
	C2	SBJ1	-	45.60 ± 7.80	54.40 ± 7.80	
		SBJ2	-	51.20 ± 4.60	48.80 ± 4.60	
	C3	SBJ1	-	69.2 ± 10.38	30.8 ± 10.38	
		SBJ2	-	62.2 ± 15.37	37.8 ± 15.37	
	O.P%					51.30 ± 2.42

Table 6.7 shows 95% confidence intervals on the average performance (mean) after k-fold (five times) cross validation tests for training data (21 subjects) as well as validation test data (9 subjects). As can be seen, both the MLP and SF-ART performed well with a few fluctuations in the classification performance. Referring to the obtained results, we didn't find any unreliability or instability in the BCG pattern recognition of 30 subjects using the SF-ART and MLP.

6.2.3 Classification of the BCG data set of 18 subjects using CombiMap+

Fig. 6.7 shows the classification results using averaged CombiMap+ for 18 subjects. As can be seen, the method is not able to recognize the heart condition of two subjects correctly, but it gives us true results for the rest of subjects. The results obtained using the CombiMap+ method was good. One person from the old unhealthy class was misclassified as healthy. This subject has unusually high J-peaks

TABLE 6.6 Effects of different MLP neural network structures and SF-ART on BCG classification performances (average) after k-fold (five times) cross validation tests using thirty subjects. C, O.P, and SBJ mean 'Class number', 'overall performance' and 'subject umber', respectively. St1: (Structure 1): two hidden layers (10, and 5 neurons, respectively); St2: two hidden layers (15, and 10 neurons, respectively); St3: two hidden layers (20, and 10 neurons, respectively); St4: three hidden layers (15, 10, and 5 neurons, respectively); the highlighted structure gave the best performance.

	Training data set (21 subjects)				Validation test data set (9 subjects)			
	C1	C2	C3	Overall	C1	C2	C3	Overall
St1	89.0%	78.0%	76.5%	84.0%	87.0%	76.5%	74.0%	81.5%
St2	91.5%	80.5%	78.5%	86.5%	89.5%	78.0%	76.0%	83.0%
St3	91.5%	80.5%	78.5%	86.5%	89.5%	78.0%	76.0%	83.0%
St.4	90.0%	79.5%	77.0%	85.0%	88.0%	77%	75.5%	82.0%
SF-ART	97.0%	78.5%	76.5%	87.0%	94.5%	76.0%	74.5%	84.5%

TABLE 6.7 95% confidence intervals on the average performance (mean) after k-fold (five times) cross validation tests. Structures: SF-ART, η and $\rho \rightarrow 1$; MLP, Nh1=15, Nh2=10, learning rate of 0.001 for all layers.

Classifier	Training data (21 subjects)	Validation test data (9 subjects)
SF-ART	87.0 ± 1.23	84.5 ± 2.67
MLP	86.5 ± 1.49	83.0 ± 2.43

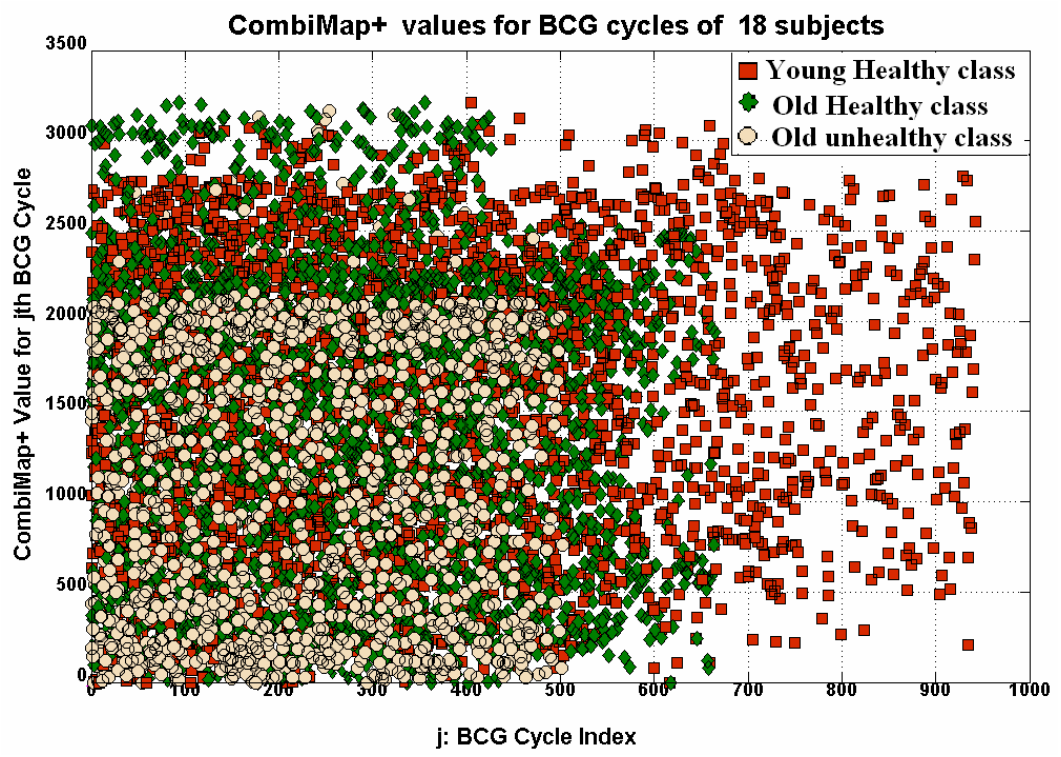
(see Fig. 2.1), which might cause the misclassification. Also, one person from the healthy class is narrowly misclassified as unhealthy. Although healthy, this person has Cardiac Index and Stroke Index values near the lower end of normal range of values. Cardiac index is the volume of blood pumped by the heart in a unit of time divided by the body surface area, usually expressed in liters per minute per square meter. Stroke volume index is ration (stroke volume)/ (body surface area). Where, stroke volume is difference between 'end diastolic volume' and 'end systolic volume' [94]. Indeed, if the person has no history of cardiac failure it does not mean that the person is fully healthy. It is possible that the person has a good recovery from infarction.

6.3 Practical applicability of research results

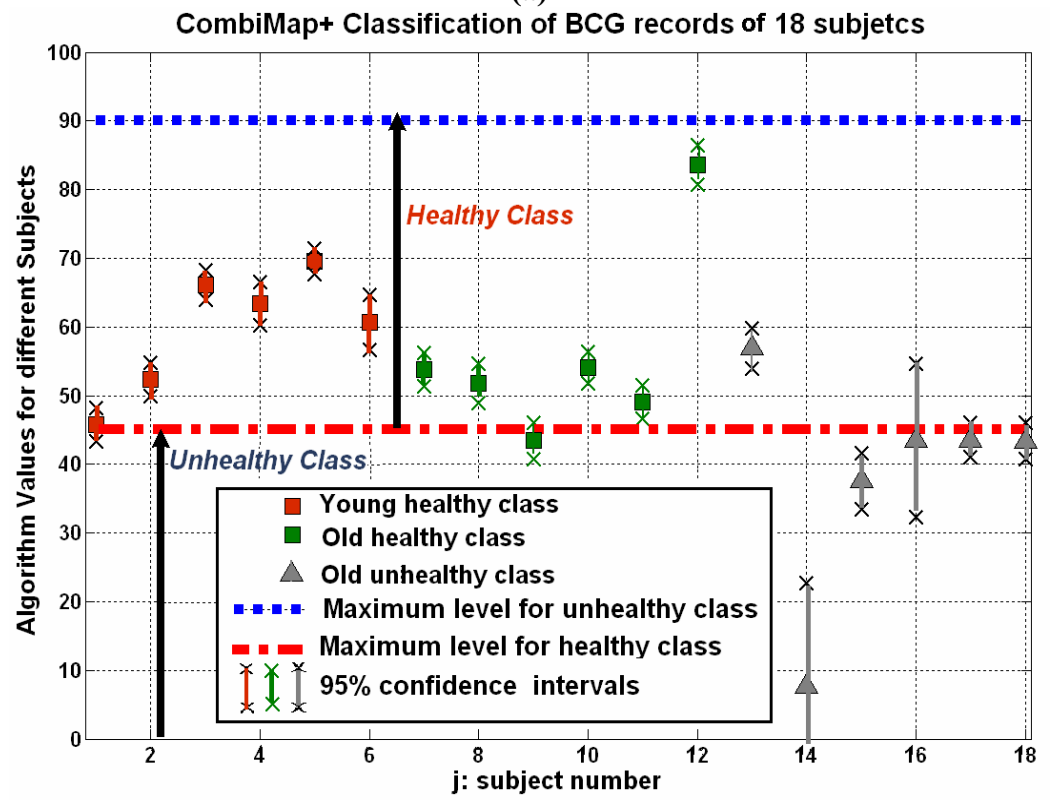
In this study we applied our newly developed learning algorithms (CombilNet, SF-ART and QuickLearn) to two well-known pattern recognition problems (image processing: IRIS with two separated and one overlapped classes; speech recognition: vowel data with three radial separated classes and one radial overlapped class); an application from real world application (BCG classification). We did not find any ineligibility for the SF-ART in these kinds of patterns but the QuickLearn had some instability for too complex patterns such as BCG. Indeed, QuickLearn still needs to be improved. However, we can apply those methods to other pattern recognition problems such as XOR, Circular classes as well as real world applications such as speech recognition, image processing, hand-written word recognition etc.

The results for our tested real data (BCG analysis) showed in last sections indicate that the three classes could be separated quite well from each other. The "scattergram" of Fig. 6.6 shows the complexity of the BCG cycles for different subjects and it suggests that the classification is not reliable on the basis of a single BCG cycle but a number of them are necessary to estimate the center of the distribution.

The accuracy of SF-ART and QuickLearn was almost the same as that of MLP, but the training time of SF-ART was much shorter. Using the same training and testing data set, the classification performance of QuickLearn was lower than with MLP but the training time was orders of magnitude shorter for all tested data (IRIS, Vowel and BCG). As we know, MLP is very sensitive to the volume of the training set as well as the number of adaptation cycles during training mode and it suffers from tradeoffs between learning speed and performance. The result showed that SF-ART and QuickLearn learnt patterns with lower volume of training samples. However, similarly to the RBF, QuickLearn suffers from weakness in local classification performance (local learning ability).



(a)



(b)

Fig. 6.7. a) CombiMap+ values for BCG Cycles of all 18 subjects (six young healthy, six old healthy, and six men with a past cardiac infarct), and b) Classification results for BCG records of these 18 subjects using shifting, scaling, and averaging CombiMap+ values, which are shown in (a), with 95% confidence intervals in the mean values.

Chapter 7

Conclusions and Future Research

7.1 Conclusions

In the first part of this work, we discussed the BCG and measurement methods. The proposed BCG signal analysis system consists of a sensitive EMFi™ film-based movement sensor, amplifiers and ADC, feature extraction and ANN classification of the features to desired classes. The EMFi™ sensor has been fitted to an office chair but it could be installed to chairs for homes or even to cars. The advantage of BCG to ECG is that no electrodes need to be attached to the subject. To extract the BCG cycles we suggested using the coarse BCG signal (raw BCG filtered by a band pass filter with 1 and 2 Hz cutting frequencies), to find the BCG cycles independently, without using ECG.

In the second part of the thesis, we applied the wavelet transform for feature extraction and well-known neural networks (MLP and RBF) for the BCG classification. The BCG waveform latency and nonlinear disturbances such as motion artifacts and electro-mechanical drifts/noises affect the BCG classification performance. Ignoring such important issues may give misleading information about patients. We used time-frequency signal processing/feature extraction methods that are compactly supported such as Daubechies and Biorthogonal wavelets, to decrease the effects of artifacts and disturbances.

To decrease the computation load during the feature extraction phase, we also introduced a novel feature extraction method called TFM-SVD. In this new method, we used statistical features of time-series and frequency series (Fourier transform of the signal). This information is then extracted into a matrix, with fixed structure from which the singular values of that matrix are sought. This transform can be used as a pre-processing stage in pattern clustering methods. The results in using it indicate that the performance of a combined system including this transform and a classifier is comparable with the performance of using other feature extraction methods such as wavelet transforms with the advantage of clearly lower computational costs. To evaluate TFM-SVD, we applied this new method and artificial neural networks for BCG classification.

In the third part of the thesis, we presented new pattern recognition methods and classifiers (QuickLearn, CombilNet and its example SF-ART) with a higher learning speed, a low computational load, higher performance, and fewer parameters to adjust

compared to classical methods. The results indicated that QuickLearn and SF-ART learned patterns (of two well known datasets, IRIS and Vowel) very fast, in less than one second for SF-ART and less than one millisecond for QuickLearn. For comparison, MLP and CNeT needed seconds or minutes to train. We also tested SF-ART and QuickLearn using BCG. The results showed clearly higher learning speed, lower computational load, easier implementation with higher or the same performance compared to present learning algorithms. Classifiers, except SF-ART and QuickLearn, were sensitive to the volume of the training set as well as the number of adaptation cycles during training mode. They also suffer from trade-offs between learning speed and performance. Moreover, the results showed that SF-ART and QuickLearn learnt patterns with a lower amount of training samples. When the shortest possible training time is important and slightly lower classification performance is allowed, QuickLearn is a very good choice. If the classification performance is most important, SF-ART can be recommended as it comes with a shorter training time than methods of equal classification performance.

In this thesis, we also developed a learning-free classifier, called averaged CombiMap+, and used it with Biorthogonal Spline wavelets to classify the BCG data of 18 subjects to healthy and unhealthy classes using a manually set threshold. Determination of this threshold automatically will be the topic of future studies.

7.2 Future Research

The development of machine learning algorithms is essential in developing intelligent systems such as autonomous robots. The European Commission sees the development of learning systems so important that it has named them one of the ICT technology pillars for the 7th Framework Program of research and development in the European Union: “Knowledge, cognitive and learning systems: capturing and exploiting knowledge embedded in web and multimedia content; bio-inspired artificial systems that perceive, understand, learn and evolve, and act autonomously; learning by machines and humans based on a better understanding of human cognition [95].”

Since future pattern recognition methods become faster and ubiquitous, there is a need for key innovations at the algorithmic level. Specifically, signal processing techniques are crucial to future learning methods. In this dissertation, we reviewed and then presented some new feature extraction and classification methods. There are still many shortages that need to be solved.

One interesting benefit of our developed learning algorithms (SF-ART and QuickLearn) is incremental learning when new data become available (on-line learning). This capability has not yet been fully explored and explained in this thesis. Therefore, a future aim can be to apply SF-ART as well as QuickLearn for classifying more subjects, especially testing its reliability, stability and performance for on-line learning. In order to extend the system into a complete “heart disease diagnostic system” several steps are necessary. First, a representative BCG data set of normal subjects of both sexes and all adult age groups needs to be collected. Additionally a representative data set of each heart disease category to be diagnosed needs to be collected and the neural network needs to be trained with the whole material. Finally, the system has to be validated with an independent set of normal

subjects and heart patients before it can be taken into clinical use. Even then the system can only serve as a part of the set of tests that confirms the diagnosis although its ease of use suggests that it could have value alone, too, as a screening device for the general population.

Although SF-ART and QuickLearn are interesting learning algorithms, and proved their abilities for pattern classification, their structures are not yet optimized and more work is necessary to optimize their structure. For instance, we can test other examples for CombilNet, generalize the SF-ART structure, and use other types of neural networks, fuzzy, neuro-fuzzy, probabilistic, or classic classifiers in the pre-classification level of the SF-ART structure to find better incremental learning algorithms. More work is also needed on QuickLearn to test different types of mapping functions to find an optimized structure. Solving the tradeoffs between learning speed, computational load, and incremental (on-line) learning abilities is not easy. Indeed, it will not be easy and it will take time to find and evaluate the optimized structures based on primary candidates (QuickLearn and SF-ART) or new potential structures which have better ability than they or other top existing methods such as Learn++ [4] and ARTMAP [60,96]. The computational load of each new method needs to be determined and their ease of implementation is compared to the present top methods. Moreover, a mathematical proof is necessary to find the stability of the final structures for QuickLearn and SF-ART, their benefits and disadvantages compared with the other existing incremental learning algorithms such as Learn++, ARTMAP etc. need to be evaluated.

Bibliography

[1] C. Stergiou and Dimitrios Siganos, Technical Report, Neural Networks, URL: http://www.doc.ic.ac.uk/~nd/surprise_96/journal/vol4/cs11/report.html.

[2] S. Grossberg, "Nonlinear neural networks: principles, mechanisms and architectures," Journal of Neural Network, Vol. 1, No. 1, 1988, pp. 17-61.

[3] E.H. Wang, and A. Kuh, "A smart algorithm for incremental learning," in Proc. of the IEEE International Joint Conference on Neural Networks (IJCNN'92), Vol. 3,1992, pp. 121-126.

[4] R. Polikar, L. Udpa, S.S. Udpa, and V. Honavar, "Learn++: an incremental learning algorithm for supervised neural networks," IEEE Transactions on Systems, Man and Cybernetics, Part C, Vol. 31, 2001, pp. 497-508.

[5] R. Polikar, J. Byorick, S. Krause, A. Marino, M Moreton, "Learn++: a classifier independent incremental learning algorithm for supervised neural networks," in Proc. of the 2002 International Joint Conference on Neural Networks (IJCNN '02), 2002, pp. 1742-1747.

[6] I. Starr, "Further clinical studies with the ballistocardiograph on abnormal form, on digitalis action, in thyroid disease, and in coronary heart disease", Trans Assoc Am Physicians, Vol. 59, 1946, pp. 180-189.

[7] B.M. Baker Jr, W.R. Scarborough and R.E. Mason, "Coronary artery disease studies by ballistocardiography: a comparison of abnormal ballistocardiograms and electrocardiograms", Ann Proc Clin. Climatol. Assoc., 1950, pp. 62-191.

[8] A. Eblen_Zajjur, "A simple Ballistocardiographic system for a medical cardiovascular physiology course", Advances in Physiology Education, Vol. 27, No. 4, December 2003, pp: 224-229.

[9] W.P.S. McKay, P.H. Gregson, B. W.S. McKay, and J. Militzer, "Sternal acceleration ballistocardiography and arterial pressure wave analysis to determine stroke volume", Clin Invest Med, Vol. 22, No. 1, 1999, pp. 4-14.

[10] Ballistocardiograph, 1953, URL: http://www.nihonkohden.com/company/history/history2.html.

[11] David M. Harrison, Ballistocardiogram, Technical report, URL: http://www.upscale.utoronto.ca/GeneralInterest/Harrison/BCG/BCG.html, Dept. of Physics, Univ. of Toronto, July 2003.

[12] I. Starr, "Prognostic value of Ballistocardiogram", JAMA, 187(7), 1964.

[13] Starr I., "Proc. First Congress Ballistocardiography and Cardiovascular Dynamics," Amsterdam, Karger, New York, 1966, pp. 7-20.

[14] E.E Eddleman, W.K. Harrison, D.H. Jackson, and H.L. Taylor, "A critical appraisal of ballistocardiography," The American Journal of Cardiology, Vol. 29, 1972, pp. 120-122.

[15] I. Starr, A. J. Rawson, H. A. Schroeder, and N. R. Joseph, "Studies on the estimation of cardiac output in man, and of abnormalities in cardiac function, from the heart's recoil and the blood's impacts", the ballistocardiogram. American Journal of Physiology, Vol. 127, 1939, pp. 1-28.

[16] I. Starr, and Henry A. Schroeder, "Ballistocardiogram.II. Normal standards, abnormalities commonly found in diseases of the heart and circulation, and their significance", Journal Clinical Invest., Vol. 19, No. 3, May 1940, pp. 437-450.

[17] T. Theorell and R. H. Rahe “Life change events, Ballistocardiography and Coronary death”, J. Human Stress, September 1975, pp. 18-24.
[18] A. Noordergraaf, and G.H. Pollack, “Ballistocardiography and Cardiac Performance”, J Bibl. cardiol. Fasc., Vol. 19, 1967, pp. 1-6.
[19] A. M. Weissler, “Noninvasive Cardiology”, Clinical cardiology monographs, Grune & Stratton Inc., New York, 1974, pp. 39-148.
[20] I. Starr, O. Horwitz, R. Mayock, and E.B. Krumbhaar, Standardization of the Ballistocardiogram by Simulation of the Heart's Function at Necropsy with a Clinical Method for the Estimation of Cardiac Strength and Normal Standards for It”, Circulation, the Journal of the American Heart Association, Dallas, TX. U.S.A., Vol. 1, No.5, May 1950, pp. 1073-1096.
[21] B.H. Jansen, B.H. Larson, K. Shankar, “Monitoring of the ballistocardiogram with the static charge sensitive bed,” Biomedical Engineering, IEEE Transactions on Vol. 38, Issue 8, Aug. 1991, pp. 748 – 751.
[22] T. Kirjavainen, O. Polo, S. McNamara, K. Vaahtoranta, and C.E. Sullivan, “Respiratory challenge induces high frequency spiking on the static charge sensitive bed (SCSB),” European Respiratory Journal, Vol. 9, 1996, pp. 1810-1815.
[23] J. Alihanka, K. Vaahtoranta, and I. Saarikivi, “A new method for long-term monitoring of the ballistocardiogram, heart rate, and respiration”, Am. J. Physiol., Vol. 240, No. 5, 1981, pp. 384-392.
[24] O. Polo, L. Brissaud, B. Sales, and A. Besset, “Billiard M., The validity of the static charge sensitive bed in detecting obstructive sleep apnoeas”, Eur Respir J, Vol. 1, 1988, pp. 330-336.
[25] E. Svanborg, H. Larsson, B. Carlsson-Nordlander, and R. Pirskanen, “A limited diagnostic investigation for obstructive sleep apnea syndrome: oximeter and static charge sensitive bed”, J Chest, Vol. 98, 1990, pp. 1341-1345.
[26] X. Yu, and D. Dent, “Neural networks in ballistocardiography (BCG) using FPGAs”, IEE Colloquium on Software Support and CAD Techniques for FPGAs, 13 Apr 1994, pp. 7/1-7/5.
[27] X. Yu, D. J. Gong, C. Osborn, and D. Dent, “A Wavelet Multireolution and Neural Network System for BCG signal Analysis”, 1996 IEEE TECON-Digital signal processing Applications., 1996, pp. 491-495.
[28] X. Yu , D. Gong , S. Li, and Yongping Xu, Evaluation of a Combined Wavelet and a Combined Principal Component Analysis Classification System for BCG Diagnostic Problem, Knowledge-Based Intelligent Information and Engineering Systems, Springer-Verlag Berlin Heidelberg, 2003, pp. 646-652.
[29] G. Zouridakis, and D. C. Tam, “Multi-unit spike discrimination using Wavelet transforms,” Elsevier Comput. Biomed., Vol. 27, No. 1, 1997, pp. 9-18.
[30] E. M. Schmidt, “Computer Separartion of multi-unit Neuroelectrical data: a review”, J. NeuroSci. Meth., Vol. 12,1984, pp. 95-111.
[31] A. Akhbardeh, A. Vahabian, and M. Farrokhi, “EEG Features Extraction using Neuro-Fuzzy Systems and Shift-Invariant Wavelet Transforms for Epileptic Seizure Diagnosing”, Proc. 26th Annual Internat. Conf. of the IEEE EMBS, San Francisco, CA, USA, 2004, pp. 498-502.
[32] J. Lekkala, and M. Paajanen, “EMFi - New Electret Material for Sensors and Actuators”, in Proc. 10th IEEE Int. Symposium on Electrets, September 22-24, 1999, pp. 743-746.

[33] S. Junnila, T. Koivistoinen, T. Kööbi, J. Niitylahti and A. Värri, “A Simple Method for Measuring and Recording Ballistocardiogram”, Proc. 17th Biennial International EURASIP Conf. Biosignal, Brno, Czech Republic, 2004, pp. 232-234.
[34] M. Koivuluoma, J. Alametsä, and A. Värri, “EMFi as Physiological Signal Sensor, First Results”, Proc. URSI XXVI convention on Radio Science and Second Finnish Wireless Communication Workshop, Tampere, Finland, 2001.
[35] D. Michael and J. Houchin, "Automatic EEG analysis: a segmentation procedure based on the autocorrelation function", Electroenceph. clin. Neurophysiol., Vol.46, 1979, pp. 232-235.
[36] A. Värri, "Digital processing of the EEG in epilepsy", Licentiate Thesis, 1988, Tampere University of Technology.
[37] H. M. Prmtorius, G. Bodenstein, and O. D. Creutzfeldt, “Adaptive segmentation of EEG records: A new approach to automatic EEG analysis”, Electruencephalogr Clin Neurophysiol, Vol. 42, 1977, pp. 84-94.
[38] U. Appel, and A. Brandt, “A comparative study of three sequential time series segmentation algorithms”, Signal Processing, North Holland, No. 6, 1984, pp. 45-60.
[39] R. Agarwal, J. Gotman, D.Flanagan and B.Rosenblatt, "Automatic EEG analysis during long-term monitoring in the ICU", Electroenceph. Clinical Neurophysiology, Vol. 107, 1998, pp. 44-58.
[40] A. Nieminen, and Y. Neuvo, “A new method for adaptive segmentation of EEG”, Proc. IV Mediterranean Conference on Med. and Bio. Eng., Sevilla, Spain, Sept. 1986, pp. 274-277.
[41] A. Glavinovitch, M.N.S. Swamy and E.I. Plotkin, “Wavelet-Based Segmentation Techniques in the Detection of Microarousals in the Sleep EEG”, 48th Midwest Symposium on Circuits and Systems, 2005., Vol. 2, 7-10 Aug. 2005, pp. 302-1305.
[42] J.K. Hiltunen, P.A. Karjalainen, and J.P. Kaipio, Comparison of the accuracy of two segmentation methods in medium rate EEG transitions, Proceedings of the 22nd Annual EMBS International Conference, Chicago IL, July 23-28, 2000.
[43] J. Hiltunen, P.A. Karjalainen, J. Partanen, and J.P. Kaipio, “Estimation of the dynamics of event-related synchronization changes in EEG”, Med Biol Eng Comput, Vol. 37, 1999, pp. 307-315.
[44] S. Mallat, A Wavelet Tour of Signal Processing (2nd ed.), Academic Press., San Diego, 1999.
[45] S. Burrus, R. Gopinoth and G. Haitao, Introduction to Wavelets and Wavelet Transforms, Printice Hall Inc., NJ, 1998.
[46] S. Haykin, “Adaptive Filter Theory”, Prentice Hall Inc., Englewood Cliffs, 1996.
[47] O. Rioul and M. Vetterli, Wavelets and signal processing, IEEE Signal Processing Magazine, Vol. 8, No. 4, Oct. 1991, pp. 14-38.
[48] A. Akhbardeh and A. Erfanian, Eye tracking User Interface using EOG signal and Neuro-Fuzzy Systems for Human-Computer Interaction aids, M.Sc. Thesis, 2001, Iran University of Science & Technology, Narmak, Tehran, Iran (Persian).
[49] P. C. Hansen, The truncated SVD as a method for regularization, BIT, vol. 27, 1987, pp. 534-553.

[50] P. C. Hansen and S. H. Jensen, “Prewhitening for Rank-Deficient Noise in Subspace,” IEEE Trans. Signal Processing, Vol. 53, No. 10, Oct. 2005.
[51] Richard O. Duda, Peter E. Hart, and D. G. Stork, Pattern classification (2nd edition), Wiley, New York, 2001.
[52] S. Haykin, Neural Networks, a Comprehensive Foundation, second ed. Prentice Hall PTR, NJ, USA, 1998.
[53] B.D. Ripley, Pattern Recognition and Neural Networks, Cambridge University Press, ISBN 0 521 46086 7, January 1996.
[54] S. Behnke, and N.B. Karayiannis, “CNeT: competitive neural trees for pattern classification,” IEEE Trans. Neural Networks, vol. 9, Nov. 1998, pp. 1352 – 1369.
[55] S. Behnke, and N.B. Karayiannis, “CNeT: competitive neural trees for pattern classification,” in Proc. of the IEEE International Conf. on Neural Networks, Vol. 3, Jun. 1996, pp. 1439-1444.
[56] V.N. Vapnik, The Nature of Statistical Learning Theory, John Wiley and Sons Interscience, U.S.A., September 16, 1998.
[57] T. Joachims, “Estimating the Generalization Performance of a SVM Efficiently,” Proceedings of the Seventeenth International Conference on Machine Learning (ICML 2000), Stanford University, Standord, CA, USA, June 29 - July 2, 2000.
[58] N. Cristianini and J. Shawe-Taylor, An Introduction to Support Vector Machines and other kernel-based learning methods, Cambridge University Press, 2000.
[59] A. Muñoz and J. M. Moguerza, “Combining Support Vector Machines and ARTMAP Architectures for Natural Classification,” Springer, Berlin, Heidelberg, Vol. 2774, 2003, pp. 16-21.
[60] G. A. Carpenter, S. Grossberg, and J. Reynolds, “ARTMAP: a self-organizing neural network architecture for fast supervised learning and pattern recognition,” in Proc. of the IEEE International Joint Conference on Neural Networks (IJCNN’91), vol. 1, 1991, pp. 863 - 868..
[61] H. G. Hosseini, K.J. Reynolds, and D. Powers, “A multi-stage neural network classifier for ECG events,” Proceedings of the 23rd Annual International Conference of the IEEE Engineering in Medicine and Biology Society, Vol. 2, Oct. 2001, pp. 1672-1675.
[62] A. Howard, C. Padgett, and C.C. Liebe, “A multi-stage neural network for automatic target detection, ” the 1998 IEEE International Joint Conference on Neural Networks Proceedings, May 1998, Vol. 1, pp. 231 - 236.
[63] H. Baltzakis, and N. Papamarkos, “A new signature verification technique based on a two-stage neural network classifier,” Journal of Engineering Applications of Artificial Intelligence, Vol. 14, Issue 1, Feb. 2001, pp. 95-103.
[64] H.C. Fu and K.P. Chiang, “Recognition of handwritten Chinese characters by multi-stage neural network classifiers,” in Proc. of the IEEE International Conference on Neural Networks, Vol. 4, 1995, pp. 2149-2153.
[65] T. Frank, K.F. Kraiss, and T. Kuhlen, “Comparative analysis of fuzzy ART and ART-2A network clustering performance,” IEEE Trans. Neural Networks, Vol. 9, May 1998, pp. 544-559.

[66] E.P Sapozhnikova, and V.P. Lunin, “A modified search procedure for the ART neural networks,” in Proc. of the 2000 International Joint Conference on Neural Networks (IJCNN '00), Vol. 5, 2000, pp. 541-544.
[67] G.A. Carpenter, and S. Grossberg, “The ART of adaptive pattern recognition by a self-organizing neural network,” IEEE Trans. Computer, Mar. 1987, pp. 77-88.
[68] K.A. Stroud and D.J. Booth, Engineering Mathematics, 5th Edition, Palgrave, Basingstoke, 2001.
[69] J. Bird, Engineering Mathematics, Newnes press, April 11, 2003.
[70] R. Kohavi, “A study of cross-validation and bootstrap for accuracy estimation and model selection,” in Proc. of the Fourteenth International Joint Conference on Artificial Intelligence, San Mateo, U.S.A, 1995, pp. 1137–1143.
[71] Confidence Intervals from statistics glossary, URL: http://www.cas.lancs.ac.uk/glossary_v1.1/confint.html.
[72] Confidence Intervals Miscellaneous on-line topics for Finite Mathematics, URL: http://people.hofstra.edu/faculty/stefan_Waner/realworld/finitetopic1/confint.html.
[73] K. Woods and K.W. Bowyer, “Generating ROC Curves for Artificial Neural Networks, IEEE Transaction on Medical Imaging,” Vol. 16, No. 3, June 1997.
[74] T. Fawcett, “ROC graphs: Notes and practical considerations for researchers,” Technical report, HP Laboratories, MS 1143, 1501 Page Mill Road, Palo Alto CA 94304, USA, April 2004.
[75] T.A. Lasko, J.G. Bhagwat, K.H. Zou and L. Ohno-Machado, The use of receiver operating characteristic curves in biomedical informatics, Journal of Biomedical Informatics 38(5), pp. 404-415, Oct. 2005.
[76] The magnificent ROC (Receiver Operating Characteristic curve), URL: http://www.anaesthetist.com/mnm/stats/roc/Findex.htm.
[77] G. Cawley, Miscellaneous MATLAB Software, Data, Tricks and Demonstrations, School of Computing Sciences, University of East Anglia, URL: http://theoval.sys.uea.ac.uk/matlab/default.html#roc.
[78] S. H. Park, J. M. Goo and Chan-Hee Jo, “Receiver Operating Characteristic (ROC) Curve: Practical Review for Radiologists,” Korean Journal of Radiology 5(1), March 2004.
[79] J.A. Hanley, and B.J. McNeil, “The Meaning and Use of the Area under a Receiver Operating Characteristic (ROC) Curve1,” Korean Journal of Radiology 5(1), March 2004.
[80] N. Lachiche and P.A. Flach, “Improving accuracy and cost of two-class and multi-class probabilistic classifiers using ROC curves,” in Proc. 20th International Conference on Machine Learning (ICML'03), AAAI Press, January 2003, pp 416-423.
[81] P. A. Flach, Tutorial on the Many Faces of ROC Analysis in Machine Learning, July 4-8, 2004, URL: http://www.cs.bris.ac.uk/~flach/ICML04tutorial/index.html.
[82] E. Anderson, “The IRISes of the Gaspe Peninsula,” Bull. Amer. IRIS Soc., Vol. 59, pp. 2–5, 1939.
[83] D. R. Hundley, Material for Neural Network course, URL: http://people.whitman.edu/~hundledr/courses/M471/iris.dat.

[84] H. C. Kim, S. Pang, H.M. Je, D. Kim and S. Y. Bang, "Support Vector Machine Ensemble with Bagging," International Workshop on Pattern Recognition with Support Vector Machines, Canada, 2002, pp. 397-407.
[85] L. Breiman, "Bagging predictors," Journal of Machine Learning, Kluwer Academic Publishers, MA,-USA, Vol. 24, Issue 2, August 1996, pp.123-140.
[86] L. K. Saul, material for Introduction to Artificial Intelligence course, URL:http://www.grasp.upenn.edu/~lsaul/teaching/cis520-fall02/hw/hw8.html.
[87] Gert Cauwenberghs, Technical Report: Fuzzy ART Classifier, 1998, URL: http://bach.ece.jhu.edu/~gert/muri/node7.html#ARTuChips.
[88] M. Cohen, P. Abshire and G. Cauwenberghs, "Mixed-Mode VLSI Implementation of Fuzzy ART," in Proc. IEEE Int. Symp. Circuits and Systems (ISCAS'98), Monterey CA, May 31-June 3, 1998.
[89] A. Akhbardeh, M. Koivuluoma, T. Koivistoinen, and A. Värri, "BCG data discrimination using daubechies compactly supported wavelet transform and neural networks towards heart disease diagnosing," Proceedings of the 2005 IEEE International Symposium on Intelligent Control and 2005 Mediterranean Conference on Control and Automation, Limassol, Cyprus, 27-29 June 2005, pp. pp. 1452-1457.
[90] A. Akhbardeh, M. Koivuluoma, T. Koivistoinen, and A. Värri, "Ballistocardiogram diagnosis using neural networks and shift-invariant daubechies wavelet transform," Proceedings of 13. European Signal Processing Conference, EUSIPCO, Antalya, Turkey, 4-8 September 2005.
[91] A. Akhbardeh, S. Junnila, M. Koivuluoma, T. Koivistoinen, and A. Värri, "The heart disease diagnosing system based on force sensitive chair's measurement, biorthogonal wavelets and neural networks," Proceedings of the 2005 IEEE/ASME International Conference on Advanced Intelligent Mechatornics, AIM 2005, Monterey, California, USA, 24-28 July 2005, pp. 676-681.
[92] A. Akhbardeh, M. Koivuluoma, T. Koivistoinen , and A. Värri, "BCG Data Clustering using New Method so-called Time-frequency Moments Singular Value Decomposition (TFM-SVD) and Artificial Neural Networks," 2005 International Symposium on Intelligent Control 13th Mediterranean Conference on Control and Automation, June 27-29, 2005, Limassol, Cyprus, pp. 1447-1451.
[93] A. Akhbardeh, S. Junnila, T. Koivistoinen, and A. Värri, "Design an Intelligent Ballistocardiographic Chair using Novel QuickLearn and SF-ART Algorithms and Biorthogonal Wavelets," 2006 IEEE International Conference on Systems, Man, and Cybernetics (SMC 2006), October 8-11, 2006, Taipei, Taiwan.
[94] M.R. Berne, and N.M. Levy, Cardiovascular Physiology (8^{th} ed.), Mosby Physiology Monograph Series, Ft Lauderdale, FL, 2001.
[95] COMMISSION OF THE EUROPEAN COMMUNITIES, Proposal for a DECISION OF THE EUROPEAN PARLIAMENT AND OF THE COUNCIL concerning the seventh framework program of the European Community for research, technological development and demonstration activities (2007 to 2013), COM. 119, Brussels, 2005, p. 23.

[96] G.A. Carpenter, S. Grossberg, “ARTMAP: Supervised real-time learning and classification of nonstationary data by a self-organizing neural network,” Neural Networks, vol. 4, 1991, pp. 565–588.

Included Publications

Publications:

Note: copy links completely and past in web-browser to access them. Otherwise it will not find the paper.

[P1] **A. Akhbardeh**, S. Junnila, M. Koivuluoma, T. Koivistoinen, V. Turjanmaa, T. Kööbi, and A. Värri, "Towards a Heart Disease Diagnosing System based on Force Sensitive chair's measurement, Biorthogonal Wavelets and Neural Network classifiers," Elsevier Journal of Engineering Applications of Artificial Intelligence (2006), http://dx.doi.org/10.1016/j.engappai.2006.07.005 (in press).

[P2] **A. Akhbardeh**, S. Junnila, M. Koivuluoma, and A. Värri, "Applying Novel Time-Frequency Moments Singular Value Decomposition (TFM-SVD) Method and Artificial Neural Networks for Ballistocardiography," EURASIP Journal on Advances in Signal Processing, accepted 10 September 2006, Vol. 2007, Hindawi Publishing Corporation, doi:10.1155/2007/60576 (in press).

[P3] **A. Akhbardeh**, and A Värri, "Applying Novel QuickLearn Algorithm for Pattern Recognition," 2006 IEEE International Conference on Systems, Man, and Cybernetics (SMC 2006), Taipei, Taiwan, 8-11 October 2006, pp. 851-855.
http://sp.cs.tut.fi/cgi-bin/cgiwrap/spwww/publication.cgi?&id=2.33/15064

[P4] **A. Akhbardeh**, and A. Värri, "Novel supervised fuzzy adaptive resonance theory (SF-ART) neural network for pattern recognition," Proceedings of IEEE International Symposium on Intelligent Signal Processing, WISP 2005, University of Algarve, Portugal, 1-3 September 2005, pp. 149-154.
http://ieeexplore.ieee.org/search/srchabstract.jsp?arnumber=1531649&isnumber=32674&punumber=10253&k2dockey=1531649@ieeecnfs&query=%28akhbardeh+a.%3Cin%3Eau%29&pos=6

[P5] **A. Akhbardeh**, S. Junnila, T. Koivistoinen, and A. Värri, "Applying Novel Supervised Fuzzy Adaptive Resonance Theory (SF-ART) neural network, Biorthogonal wavelets for Ballistocardiogram diagnosis," 2006 IEEE Conference on Control Applications, , Munich, Germany, 4-6 October 2006, pp. 143-148.
http://ieeexplore.ieee.org/search/srchabstract.jsp?arnumber=4064917&isnumber=4064892&punumber=4064891&k2dockey=4064917@ieeecnfs&query=%28akhbardeh+a.%3Cin%3Eau%29&pos=1

[P6] **A. Akhbardeh**, S. Junnila, T. Koivistoinen, and A. Värri, "Applying Biorthogonal wavelets and a Novel QuickLearn Algorithm for an Intelligent Ballistocardiographic chair," 2006 IEEE Mountain Workshop on Adaptive and Learning Systems (SMCals'06), Utah State University, College of Engineering, Logan, U.S.A., 24–26 July 2006, pp. 42-47.
http://ieeexplore.ieee.org/search/srchabstract.jsp?arnumber=4016760&isnumber=4016752&punumber=4016751&k2dockey=4016760@ieeecnfs&query=%28akhbardeh+a.%3Cin%3Eau%29&pos=4

[P7] **A. Akhbardeh**, S. Junnila, M. Koivuluoma, T. Koivistoinen, and A. Värri, Evaluation of heart condition based on ballistocardiogram classification using compactly supported wavelet transforms and neural networks, Proceedings of the 2005 IEEE Conference on Control Applications, Toronto, Canada, 29-31 August 2005, pp. 843-848.

http://ieeexplore.ieee.org/search/srchabstract.jsp?arnumber=1507234&isnumber=32294&punumber=10069&k2dockey=1507234@ieeecnfs&query=%28akhbardeh+a.%3Cin%3Eau%29&pos=10

[P8] **A. Akhbardeh**, S. Junnila, T. Koivistoinen, T. Kööbi, and A. Värri, "Ballistocardiogram classification using a novel transform so-called AliMap and biorthogonal wavelets," Proceedings of IEEE International Symposium on Intelligent Signal Processing, WISP 2005, University of Algarve, Portugal, 1-3 September 2005, pp. 64-69.
http://ieeexplore.ieee.org/search/srchabstract.jsp?arnumber=1531649&isnumber=32674&punumber=10253&k2dockey=1531649@ieeecnfs&query=%28akhbardeh+a.%3Cin%3Eau%29&pos=6

[P9] S. Junnila, **A. Akhbardeh**, A. Värri, and T. Koivistoinen, "An EMFi-film sensor based ballistocardiographic chair: performance and cycle extraction method," Proceedings of 2005 IEEE Workshop on Signal Processing Systems, Design and Implementation, SIPS 2005, Athens, Greece, 2-4 November 2005, pp. 373-377.
http://ieeexplore.ieee.org/search/srchabstract.jsp?arnumber=1579896&isnumber=33371&punumber=10549&k2dockey=1579896@ieeecnfs&query=%28akhbardeh+a.%3Cin%3Eau%29&pos=5

[P10] S. Junnila, **A. Akhbardeh**, L. C Barna, Irek Defee, and A. Värri, "A Wireless Ballistocardiographic Chair," 28th Annual International Conference IEEE Engineering in Medicine and Biology Society, Aug.30-Sept.3, 2006, New York City, New York, USA, pp. 5932-5935.
http://ieeexplore.ieee.org/search/srchabstract.jsp?arnumber=4029514&isnumber=4028926&punumber=4028925&k2dockey=4029514@ieeecnfs&query=%28akhbardeh+a.%3Cin%3Eau%29&pos=0

Tampereen teknillinen yliopisto
PL 527
33101 Tampere

Tampere University of Technology
P.O. Box 527
FIN-33101 Tampere, Finland