



YAŞAR UNIVERSITY
GRADUATE SCHOOL OF NATURAL AND APPLIED SCIENCES

MASTER THESIS

**DESIGN AND IMPLEMENTATION OF ECG BASED
WEARABLE FITNESS TRACKER**

ÇAĞLA SARVAN

THESIS ADVISOR: DR. NALAN ÖZKURT

ELECTRICAL AND ELECTRONICS ENGINEERING

PRESENTATION DATE: 29.08.2018


BORNOVA / İZMİR
AUGUST 2018

We certify that, as the jury, we have read this thesis and that in our opinion it is fully adequate, in scope and in quality, as a thesis for the degree of Master of Science.

Jury Members:

Asst. Prof. Dr. Nalan ÖZKURT
Yaşar University

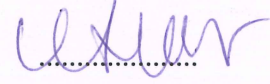
Signature:

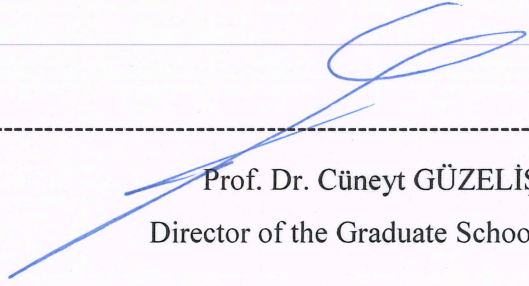


Asst. Prof. Dr. Korhan KARABULUT
Yaşar University



Prof. Dr. Mehmet ENGİN
Ege University




Prof. Dr. Cüneyt GÜZELİŞ
Director of the Graduate School

ABSTRACT

DESIGN AND IMPLEMENTATION OF ECG BASED WEARABLE FITNESS TRACKER

Sarvan, Çağla

Msc, Electrical Electronics Engineering

Advisor: Dr. Nalan Özkurt

August 2018

In this thesis, it is aimed to design a wearable device which can follow up activities via real time heart beats such as sitting, walking and running; and also aimed to obtain highly discriminative features of normal, right bundle branch block (RBBB), left bundle branch block (LBBB) and paced heart rhythms which were downloaded from MIT-BIH database.

Arrhythmia classification by using discrete wavelet transform (DWT) features can be considered in three different parts. In the first study, four arrhythmias were classified with neural network (NN) by 16 statistical features extracted from DWT coefficients of a single wavelet and 32 features extracted from two different wavelets. It has been observed that, the most successful 32 Features extracted from two wavelets among others performed greater classification accuracy compared to a single wavelet.

Then, a genetic algorithm (GA) has been used in order to classify arrhythmia beats using high accuracy discriminative features which have been investigated using various wavelets having combinations of detail coefficients within different levels. In this proposed DWT method, a pool has been formed using features of various wavelets. The 16 features has been chosen by GA which exhibited the greatest accuracy rate. The accuracy rate of the NN has been used as the fitness function and is tried to be maximized.

Multi-objective GA (MOGA) is also another approach to select suitable features for detecting heart signal arrhythmia types characteristics clearly. In this approach, selection has been done by reduction of criteria to one scale within the feature pool. Accuracy rate and root mean square error of NN and number of features have been used in the process of fitness function evaluation. Smallest feature set, with highest

accuracy rate and least error has been searched by using three criteria instead of one, by a GA with multi-objective approach.

A simulation has been conducted to test features obtained by different methods. Multi-layer perceptron (MLP) training outputs of; 32 features from NN, 16 features selected from GA and 48 features chosen by GA with multi-objective approach have been studied on. Arrhythmia type was identified with trained MLP by selecting one of the specified feature sets using the developed graphical user interface.

A compact circuit design has been implemented in order to investigate real-time heart beats. The movement data and related heart beat are acquired by an accelerometer and the signals are transmitted with a bluetooth module via a microprocessor to interface program wirelessly.

Key Words: Heart Beat Classification of ECG, arrhythmia, discrete wavelet transform, wavelet features, feature selection, neural network, genetic algorithm, multi-objective optimization, real-time heart signals, accelerometer, wearable ECG device



ÖZ

EKG TABANLI GİYİLEBİLİR AKTİVİTE TAKİP CİHAZININ TASARIMI VE UYGULANMASI

Sarvan, Çağla

Yüksek Lisans Tezi, Elektrik ve Elektronik Mühendisliği

Danışman: Dr. Nalan ÖZKURT

Ağustos 2018

Bu çalışmada MIT-BIH veri tabanından elde edilen normal, sağ dal bloğu, sol dal bloğu ve pace ritimlerine ait kalp vuruları verisi kullanılarak ayrıştırıcılığı yüksek özniteliklerin bulunması ve kalp sinyallerinin gerçek zamanlı alınmasıyla, aktivite takibi yapabilen, giyilebilir bir cihaz tasarımı hedeflenmiştir.

Kesikli dalgacık dönüşümü (KDD) öznitelikleri ile aritmi sınıflandırma çalışması üç kısımda incelenebilir. İlk çalışmada dört aritmi tipi sinir ağları ile tek bir dalgacıktan elde edilen 16 istatistiksel öznitelikle ve iki farklı dalgacıktan elde edilen 32 öznitelikle sınıflandırılmıştır. Sınıflandırmada en başarılı bulunan iki dalgacıktan elde edilen 32 adetlik özniteliğin tek dalgacığa göre daha yüksek başarımlı oranı sağladığı gözlemlenmiştir.

Daha sonra, incelenen aritmi vurularının sınıflandırılmasında farklı tip dalgacıkların farklı seviyede seçilen detay katsayılarının kombinasyonu ile yüksek başarımlı oranı sağlayan özniteliklerin seçilimi için genetik algoritma yönteminden faydalanılmıştır. Önerilen KDD yöntemi ile farklı tip dalgacıklardan bir öznitelik havuzu oluşturulmuştur. Sınıflandırma algoritmasında yüksek doğruluk oranı veren 16 adetlik öznitelik seti genetik algoritma yöntemi ile seçilmiştir. Uygunluk fonksiyonunda sinir ağının doğruluk oranı uygunluk fonksiyonu olarak kullanılmıştır ve bu değer maksimize edilmeye çalışılmıştır.

Kalp sinyali aritmi tiplerinin karakteristiğini yansıtan uygun özniteliklerin tespit edilebilmesi için önerilen diğer bir yöntem çok amaçlı yaklaşımla GA kullanılmasıdır. Bu yöntemde oluşturulan öznitelik havuzundan seçim birden fazla kriterin tek bir değere indirgenmesiyle yapılmıştır. Uygunluk fonksiyonu değerlendirmesinde sinir ağı sınıflandırmasının doğruluk oranı, ortalama karekök hata oranı ve seçilen öznitelik sayısı kullanılmıştır. Yüksek doğruluk oranını az hata ile sağlayan en düşük sayıda

öznitelik seti aranarak bir yerine üç adet kriter çok amaçlı yaklaşımla GA kullanılarak değerlendirilmiştir.

Farklı yöntemler ile tespit edilen öznitelik setlerinin testi için bir simülasyon programı oluşturulmuştur. Simülasyon programında iki adet dalgacığın birleşiminden elde edilen 32 adetlik öznitelik seti, GA tarafından seçilen 16 adetlik öznitelik seti ve çok amaçlı yaklaşımla GA kullanılarak tespit edilen 48 adetlik öznitelik setinin çok katmanlı algılayıcı (MLP) eğitim çıktısı kullanılmıştır. Geliştirilen grafik arayüzü ile seçilen öznitelik setini kullanarak MLP aritmi tipini belirler.

Gerçek zamanlı kalp sinyallerinin elde edilmesi için kompakt bir devre tasarımı yapılmıştır. Kalp sinyalleri ile birlikte eş zamanlı hareket bilgisinin alınması için ivmeölçer kullanılmış olup sinyaller bir mikrodenetleyici üzerinden arayüz programına bluetooth modülüyle kablosuz olarak aktarılmıştır. Sinyaller tasarlanan arayüz programında gerçek zamanlı olarak çizdirilmiştir. Tasarlanan devre kartı 3D yazıcı ile basılan kompakt bir kutu içerisine yerleştirip giyilebilir bir EKG cihazı haline getirilmiştir. Böylelikle aktivite sırasında oluşan kalp ritimleri gerçek zamanlı olarak gözlemlenip kayıt altına alınabilmektedir.

Anahtar Kelimeler: EKG vuru sınıflandırılması, aritmi, kesikli dalgacık dönüşümü, dalgacık öznitelikleri, öznitelik seçimi, yapay sinir ağları, genetik algoritmalar, çok amaçlı eniyileme, gerçek zamanlı kalp sinyalleri, ivmeölçer, giyilebilir EKG cihazı

ACKNOWLEDGEMENTS

I would first like to thank my thesis advisor Dr. Nalan Özkurt who taught me the rules of academic studies. Her attitude of coming up with new ideas eventually infected me in an academically productive way. The curiosity she showed during searching and determination while proceeding was more than encouraging. I wouldn't have completed this thesis without her scientific and parental guidance. When I had even the smallest hesitations while implementing my ideas I felt free to ask her by e-mail or WhatsApp even on Sundays. She steered me in the right way while studying and even more she supported me mentally whenever she thought I needed it. All her contributions will be appreciated for forever.

Secondly, I would like to express my gratitude to Dr. Korhan Karabulut for teaching and guidance on evolutionary algorithms as a technique that I used within this thesis.

I want to express my appreciation to my family who raised me, who are my shelter, my unconditional supporters, and my safest place. I will show the same effort and even more in the rest of my life against them. I will be as delicate as they have been when I was a kid and love them the way they caressed me.

I would like to verbalize my deepest thanks to my dearest boyfriend Erinç Cibil. He stood by me during this whole process and showed his best patience. He is a talented software engineer hence we performed brainstorming numerous times. My sincere gratitude goes for him and all his efforts during my MSc. thesis will be remembered.

I would like to thank, especially Bilgi Özkan, Research Asst. Ceyhan Türkmen , Barbaros Küçük and Cihangir Şimşekcan for their contributions during hardware implementation and boxing.

And finally, I would like to thank my friend Research Asst. Zeynep Ertekin for her mental and spiritual supports. She was practically my second eye during writing process, her contributions have a particular importance in this study.

This study would not have been successfully delivered without the names given above. Thank you.

Çağla SARVAN

İzmir, 2018

TEXT OF OATH

I declare and honestly confirm that my study, titled “DESIGN AND IMPLEMENTATION OF ECG BASED WEARABLE FITNESS TRACKER” and presented as a Master’s Thesis, has been written without applying to any assistance inconsistent with scientific ethics and traditions. I declare, to the best of my knowledge and belief, that all content and ideas drawn directly or indirectly from external sources are indicated in the text and listed in the list of references.

Çağla Sarvan

Signature



September 4, 2018

TABLE OF CONTENTS

ABSTRACT.....	v
ÖZ.....	vii
ACKNOWLEDGEMENTS.....	ix
TEXT OF OATH.....	xi
TABLE OF CONTENTS.....	xiii
LIST OF FIGURES.....	xv
LIST OF TABLES.....	xix
CHAPTER 1 SCOPE OF THESIS AND LITERATURE REVIEW.....	1
1.1. The Motivation and Literature Review.....	1
1.2. Aim of Study.....	4
1.3. Outline of Thesis.....	5
CHAPTER 2 HEART ANATOMY.....	7
2.1. Heart.....	7
2.2. Electrocardiography.....	10
2.3. Cardiac Arrhythmias.....	10
2.3.1. Normal.....	11
2.3.2. Left Bundle Branch Block.....	12
2.3.3. Right Bundle Branch Block.....	13
2.3.4. Paced.....	14
CHAPTER 3 SIGNAL PROCESSING AND MACHINE LEARNING.....	15
3.1. Wavelet Transform.....	15
3.1.1. Continuous Wavelet Transform.....	15
3.1.2. Discrete Wavelet Transform.....	18
3.2. Neural Networks.....	21
3.2.1. Multi-Layer Perceptron.....	26
3.3. Evolutionary Algorithms.....	28
3.3.1. Genetic Algorithm.....	30
3.3.2. Multi-Objective Genetic Algorithm.....	33
CHAPTER 4 ARRHYTMIA DETECTION.....	35
4.1. Data Set.....	35
4.2. Pan Tompkins Algorithm for QRS Detection.....	36
4.3. Features and Feature Extraction.....	44
4.4. Feature Selection.....	52
4.4.1. GA Implementation with Matlab.....	52
4.4.2. Multi-Objective GA Implementation with JAVA.....	59

4.5. Summary of Arrhythmia Detection.....	66
4.6. JAVA Implementation of Arrhythmia Detection.....	66
4.7. Discussion	71
CHAPTER 5 ACTIVITY MONITORING WITH MOBILE ECG	75
5.1. Hardware	75
5.1.1. BMD101	75
5.1.2. Accelerometer.....	77
5.1.3. Bluetooth Module	79
5.1.4. Microcontroller	80
5.1.5. Circuit Scheme.....	81
5.2. Software	83
5.3. Results and Discussion.....	88
CHAPTER 6 CONCLUSION AND FUTURE STUDIES	92
REFERENCES	97
APPENDIX 1 –Decomposition Filter Coefficients of Wavelet.....	101

LIST OF FIGURES

Figure 2.1. Schematic of Cardiac Cycle	7
Figure 2.2. P-QRS-T Signal Shape	9
Figure 2.3. Normal Heart Beat.....	12
Figure 2.4. Heart Beat of LBBB Rhythm	13
Figure 2.5. Heart Beat of RBBB Rhythm	13
Figure 2.6. Heart Beats of Paced Arrhythmia	14
Figure 3.1. Wavelet Analysis of a Signal.....	17
Figure 3.2. A Schematic Representation of Signal $x[n]$	19
Figure 3.3. Computation of DWT for 5 Scale Index.....	20
Figure 3.4. Schematic of Filtering Process	21
Figure 3.5. Representation of a Nervous System.....	22
Figure 3.6. Diagram of Non-linear Neuron.....	22
Figure 3.7. Neural Network Models	23
Figure 3.8. Recurrent Neural Network Representation.....	24
Figure 3.9. Block Diagram Of Neural Network Classification.....	25
Figure 3.10. Multi Layer Perceptron Scheme with Two Hidden Layers	26
Figure 3.11. Basic Parts of EA.....	29
Figure 3.12. Block Diagram of Phenotype Corresponding to the Genotype	30
Figure 3.13. One point and Two Point Crossover Representation.....	32
Figure 3.14. Uniform Crossover Representation	32
Figure 4.1. Block Diagram of Pan Tompkins Algorithm.....	36
Figure 4.2. Application of Pan Tompkins Algorithm on ECG Signal	40
Figure 4.3. Finding R-peaks of an ECG Signal	40
Figure 4.4. Detection and Extraction of QRS	41
Figure 4.5. Four level DWT Decomposition of A Normal Heart Beat.....	46

Figure 4.6. Block Diagram of Feature Extraction	47
Figure 4.7. Comparison Bar Graph of Training Functions Accuracy Results.....	49
Figure 4.8. Bar Graph of Different Wavelet Type's Accuracy Rates.....	50
Figure 4.9. Genotype and Phenotype Representation of GA.....	53
Figure 4.10. Architecture of GA.....	56
Figure 4.11. Graph of Fitness Function Values of Generations	58
Figure 4.12. Architecture of GA with Multi-Objective Approach	63
Figure 4.13. Fitness Function with Respect to Generations	64
Figure 4.14. Multi-Objective Approach Criterias with Respect to Generations.....	65
Figure 4.15. Selection of Feature Set from Combobox	69
Figure 4.16. Selection of ECG Signal Type	69
Figure 4.17. Simulating Signal and Labelling the Beats	70
Figure 4.18. Pop-up Window for Beat Analysis in Multiwavelet	70
Figure 4.19. Pop-up Window for Beat Analysis in MOGA	71
Figure 5.1. Block Diagram of AFE	76
Figure 5.2. BMD101 Packet Form	76
Figure 5.3. Pin Connection Diagram of BMD101.....	77
Figure 5.4. Axes and Direction Scheme of GY-521.....	78
Figure 5.5. HC-05 Scheme	79
Figure 5.6. Pin Layout of Arduino Nano.....	80
Figure 5.7. Circuit Scheme of ECG Device	81
Figure 5.8. Application Schematic of Wearable ECG Device	82
Figure 5.9. Photo of the ECG Device	83
Figure 5.10. Data Row Format of the BMD101	84
Figure 5.11. Parsed Value Packet of BMD101.....	86
Figure 5.12. Measured Value Packet of GY521	87
Figure 5.13. Serial Data Packet of ECG Device.....	87

Figure 5.14. Heart Signal and Accelerometer Signal of Sitting Person..... 88

Figure 5.15. Heart Signal and Accelerometer Signal of Walking Person..... 89

Figure 5.16. Heart Signal and Accelerometer Signal of Running Person..... 89

Figure 5.17. Illustration of Heart Rate Calculation..... 90



LIST OF TABLES

Table 2.1. Heart Beat Classes	11
Table 4.1. MIT-BIH Records with Details.....	36
Table 4.2. Wavelet Types.....	46
Table 4.3. Accuracy Rates of Classification of Wavelet Types for Arrhythmias	50
Table 4.4. Neural Network Classification Results of Feature Sets	52
Table 4.5. GA Results of Three Different Run	59
Table 4.6. Parameters of MLP for MOGA.....	60
Table 4.7. Feature Set of Three Methods.....	67
Table 4.8. Simulation Data Detail of the Software	68
Table 4.9. Simulation Results of Normal Rhythm Records Classification	72
Table 4.10. Simulation Results of LBBB Records Classification.....	73
Table 4.11. Simulation Results of RBBB Records Classification	73
Table 4.12. Simulation Results of PACED Records Classification.....	74
Table 5.1. Configuration Register Map of GY-521	78
Table 5.2. AFS_SEL Bits Configuration Values	78
Table 5.3. Pin Descriptions of Arduino Nano.....	81
Table 5.4. Byte Descriptions of Payload Array	85

SYMBOLS AND ABBREVIATIONS

ABBREVIATIONS:

SA	Sino Atrial
HR	Heart Rate
AV	Atrioventricular
RA	Right Atrium
RV	Right Ventricle
LA	Left Atrium
LV	Left Ventricle
ECG	Electrocardiogram
FT	Fourier Transform
STFT	Short Time Fourier Transform
WT	Wavelet Transform
CWT	Continuous Wavelet Transform
NN	Neural Network
GA	Genetic Algorithm
MLP	Multi-Layer Perceptron
MOGA	Multi-Objective Genetic Algorithm
DWT	Discrete Wavelet Transform
LBBS	Left Bundle Branch Block
RBBS	Right Bundle Branch Block
HPF	High Pass Filter
LPF	Low Pass Filter
LNA	Low Noise Amplifier
ADC	Analog Digital Converter

RMSE Root Mean Square Error

GUI Graphical User Interface



CHAPTER 1

SCOPE OF THESIS AND LITERATURE REVIEW

1.1. The Motivation and Literature Review

Human health is an important issue in terms of correct operation of vital functions. In this respect, heart is located at the center of human body which is a major health concern. The correct diagnosis of disorders in heart acts a serious role while treating patients quickly.

To understand the working principle of human body, the electrical signals produced by the human body must be examined. These electrical signals are also called as bio-signals and they can be measured or observed by using medical devices. Bio-signals help to observe a person's vital functions. High-precision medical measurement instruments are being developed to analyze these signals. One of these devices is electrocardiogram (ECG) which reflects the heart signal characteristic during a cardiac cycle. By examination of a cardiac cycle, the disorders in the heart can be identified by specialists. Unusual rhythms arisen during a heart beat is called as an arrhythmia. Different arrhythmia types cause different heart signal characteristics. These characteristics can imply hints on the disease knowledge's to experts. In this context, developing semi-automatic systems that assist experts is an investigated issue. Differentiation of a healthy individual and a person with a disease is the main part of that concept.

Signal processing methods are used in the research of heart signal characteristics. The researchers proposed Shannon energy in order to find out QRS complexes in ECG signal. The algorithm they have used can detect QRS complexes less than 0.013 seconds, almost without error (Beyramienanlou and Lotfivand, 2017). A research team proposed a novel approach in the detection and classification process of some kind of vital arrhythmia; ventricular arrhythmia. In their research the ECG signal has been analyzed by digital Taylor Fourier transform (DTFT) and phase feature has been considered newly (Triphaty et al., 2018). Gonçalves and his co-workers (2013)

compared Short Time Fourier Transform (STFT) and Continuous Wavelet Transform (CWT) while detecting and localizing embolic blood flow signals. Superiority of CWT over STFT has been shown clearly by using six different type of wavelets.

Feature extraction from the heart signal is the most commonly used structure that helps to categorize heart beats according to their characteristics. In morphological methods the features are extracted from the heart signal by revealing mathematical model of the heart signal structure with the time intervals between the electrical points which are significant in heart beat (De Chazal et al., 2004; Mazomenos et al., 2012; Ince et al., 2009) or finding best PQRST fragments by using Time domain approach for biometric application (Patro and Kumar, 2017). Another approach is separating signals that include multi-components to sub-components with Independent component analysis (Yu and Chou, 2008; Jiang et al., 2006). Recently, obtaining frequency coefficients with restricted time interval from the signal by using Fast Fourier Transform is a widely used technique in bio-medical signals (Mironovova and Bíla, 2015). Wavelet Transform which performs localization analysis for non-stationary signals like ECG by providing information both in time and frequency domain is a highly discriminative signal processing application technique (Uslu and Bilgin, 2008; Shufni and Mashor, 2015). By researchers, extracting features from a bio-medical signal such as an ECG signal by using DWT is mostly preferred method due to its computationally efficient algorithm (Sarkaleh and Shahbahrami, 2012). The important issue in DWT is to choose the best fitting wavelet type according to the signal shape. Because if the chosen wavelet is suitable for the signal, the better features are extracted from it. Castro and his friends (2010) proposed feature extraction by using suitable mother wavelet.

To understand which method or features reflect an ECG signal best, the accuracy rate of classification algorithms is taken as reference. There are different types of classification approaches such as deep neural network offered by Sanino and Pietro (2018) which categorizes ECG arrhythmias automatically. Multi Layer Perceptron is another neural network structure that uses features for classification (Plawiak, 2018).

The researchers proposed Mahalanobis and Minimum distance based classifiers by using wavelet entropy and AR coefficients, which are statistical features, in their ECG beat classification study. They performed feature extraction from normal, LBBB, RBBB and paced arrhythmias that obtained from the MIT-BIH database. In their classifier comparison, the Mahalanobis distance based classifier ensures accuracy over

90% (Engin et al., 2007).

The feature selection technique of genetic algorithm (GA) was offered to find the most suitable feature sets. The selected feature subsets from GA can be effective in classification algorithm's accuracy rate (Huang and Wang, 2006).

A researcher investigated 744 ECG signals of 29 patients, comparing machine learning methods and reached 98.99% classification accuracy of 17 heart diseases by applying genetic optimization of classifiers parameters (Plawiak, 2018). Also Yang and Honavar (1998) are briefly explaining the reasons of using GA for feature selection in their study.

GA is mostly used during optimization of problems such as finding solutions for credit risk assessment (Oreski and Oreski, 2014) or searching variables for a partial least squares model (Leardi and Gonzalez, 1998). When the fragments of the problem are required to be investigated profoundly, solutions are searched by using multi-objective GA.

Fonesca and Fleming (1993) give brief information about multi-objective selection approach by changing fitness assignment of GA and express that using more than one criterion in order to find solutions to real world problems, could produce better solutions.

Címpanu and friends (2017) applied a multi-objective GA for optimization of accuracy by selecting required features for classification algorithms; Support Vector Machine and Random Forests. In their research, Multi-objective approach found more efficient features than single objective approach of GA structure for EEG signal classification. Using Multi-objective approach in GA provides unlimited balance between objectives of the classifier.

The studies mentioned above all indicated the lack of real-time ECG data acquisition instrument especially for children with chronic illnesses and for the elder, therefore scientists headed towards to design of a wearable real-time ECG device. Sun and his research team (2017) from China, designed a lightweight wearable Health-shirt (H-shirt) which provides an ECG monitoring system. In addition, heart rate based lactate threshold has been realized where it implies the lactic acid, a well-known enzyme, forms during and after exercise.

Researchers developed a remote ECG monitoring system where Bluetooth sensor delivers person's data on a mobile phone. The GPRS module attached to system ensures up-to-date condition of the heart disorder. By such a system, straight and reliable communication between patient and a doctor has been aimed (Hussain et al., 2016).

Another wearable belt type ECG monitoring device, which can also detect the status of the activity, has been implemented by a researcher group in South Korea. They put forward an efficient approach to remove the artifacts formed during motion, Empirical Mode Decomposition (EMD) filtering which improved R-peak detection significantly (Kim et al., 2015).

Today, developments in digital world are almost out of reach where computer and electronics scientists work together. The common study of these two disciplines yields cutting-edge technology products. Even sometimes, medical scientists get involved with these studies. A research group from United States has developed a cloud based real-time ECG monitoring system. Even though the system is multi objective, they aimed to collect, store and transmit ECG data automatically to medical experts at anytime and anywhere. The related feedback can be turned back to patients via same network. In this study, ECG segmentation has been realized by Wavelet Transform and cloud-software design has been implemented on Amazon web service and related database which includes both doctor and patient (Xia et al., 2013).

By taking the literature review and verified studies into consideration, implementation of a wearable ECG device and ECG rhythms classification have been aimed this study.

1.2. Aim of Study

Nowadays, health studies mostly concentrate on systems with high sensitivity that can provide fast and accurate results. When the issue is human health, heart is an important place in the literature. The systems are being developed that can help the specialists to diagnose heart diseases day to day. This thesis have two main objectives: The first one is to analyze the cardiac arrhythmias and to construct an efficient feature set for arrhythmia classification. The second purpose of this thesis is to measure, display and record real-time heart beats with a comfortable wearable device by showing the current activity status.

Heart beat that is obtained from a medical device can give a visual information to specialists. A specialist can diagnose the disease according to the heart signal pattern through an ECG device. However, in heart disorders, the patient's heart signals should be monitored or recorded for 24 hours that means many beats and each beat may not contain an arrhythmia. Detecting and finding arrhythmia from records or from real-time systems with an automatic or semi-automatic technique not only saves time for experts, but also ensure the possibility of quick diagnoses and initiation of the treatment period simultaneously. In this thesis, it is aimed to develop a technique which can detect arrhythmias with high accuracy and a product which performs activity-related observation on real-time heart signals.

To categorize heart signals according to arrhythmia types, a feature extraction method have been developed by using signal processing tools. Time and frequency information of ECG signal was obtained and the features were constituted from calculating statistical values such as mean, standard-deviation, entropy. The most suitable features were selected by using machine learning techniques. Classifying heart beats with high accuracy rate in neural network classification with selected features set. Genetic Algorithm (GA) and GA with multi-objective approach is proposed in this thesis as the feature selection method. The most suitable feature set was found for detection of four arrhythmia types and the simulation program was designed to test these features.

The compact and wearable ECG device was designed with an accelerometer. Using Bluetooth, information was transferred to the computer with wireless communication. Real-time heart signals and activity information were examined through the developed interface program.

As a result, it was observed that arrhythmia can be detected through recorded signals with high accuracy rate and real-time heartbeats can be observed through an interface software with activity information.

1.3. Outline of Thesis

Chapter 1 includes literature studies related to this thesis. The thesis was shaped according to these researches. Chapter also conveyed the aim of creating this thesis.

In Chapter 2 gives brief information about heart structure. It describes heart's working principles, electrocardiography device by which heart signal measurements are

recorded and includes information about cardiac arrhythmias that are used in this thesis.

Chapter 3 includes a signal processing tool of Wavelet Transform (WT) and background theory of Neural Networks (NN) and Evolutionary Algorithms (EA).

Chapter 4 contains features that are extracted with the technique of DWT, selected feature sets from Genetic Algorithm (GA) and GA with multi-objective approaches. Test results of the classification are shown in a simulation software.

In Chapter 5, the design and implementation of wearable device is considered in both hardware and software. Then, the test results are demonstrated.

In Chapter 6, the contributions of thesis have been discussed. According to the results, future works and expectations have been described.

CHAPTER 2

HEART ANATOMY

2.1. Heart

Heart is an electrical, closed fist sized pump which is in charge of generating and commanding signals to induce a heartbeat. These electrical pulses trigger the heart's muscle to contract. By contraction, the process of bloodstream starts.

There exist four chambers in the heart, upper two chambers atria; receive and store deoxygenated (venous) blood whereas lower two chambers ventricles are responsible for pumping oxygenated blood to body and lungs through the circulatory system. Every individual has his own natural (or artificial) master pacemaker cells which form the sino-atrial (SA) node. The heart rate (HR) is managed by this specific node (Rangayyan, 2015) .

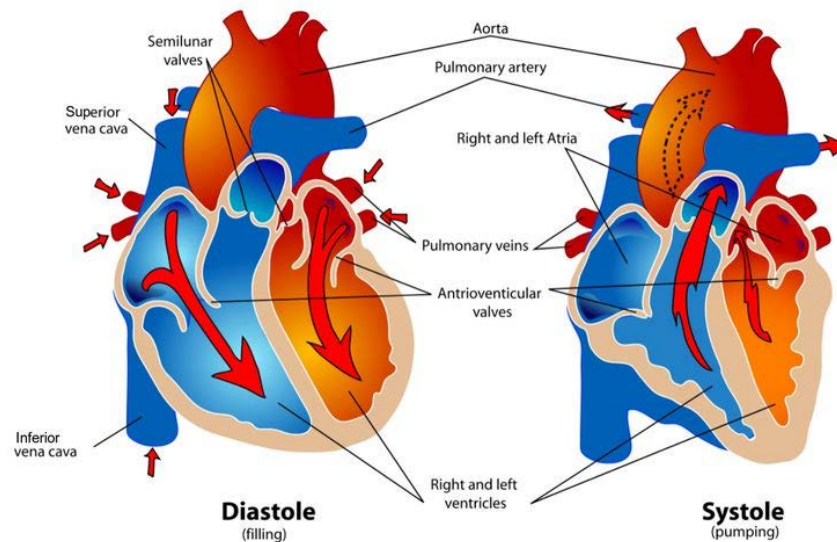


Figure 2.1. Schematic of Cardiac Cycle (From Bailey 2018)

The electrical system of heart consists of three main parts:

- The sinoatrial (SA) node is in the right atrium

- The atrioventricular (AV) node is on the interatrial septum
- The His-Purkinje fibres are throughout the walls the heart's ventricles

The right atrium (RA) receives deoxygenated blood from superior and inferior vena cava which are main venous blood vessels incoming to heart. Via tricuspid valve blood is transmitted from the RA to the right ventricle (RV) through atrial contraction. In ventricular systole, the deoxygenated blood in the RV is pumped out by the pulmonary valve towards the lung for cleansing (oxygenation). The contraction of the heart in order to pump the blood out is named as Systole whereas the relaxation after contraction is defined as diastole (Figure 2.1).

The left atrium (LA) delivers arterial blood from the lungs, which is transmitted to the left ventricle (LV) through mitral valve in atrial contraction. The biggest and most significant cardiac chamber is LV since it contracts the strongest among all others in order to pump out the purified blood via the aortic valve encountering the pressure of the remaining part of blood vascular system (Rangayyan, 2015).

Neurotransmitters acetylcholine or epinephrine are controlled by signals from autonomous and central nervous systems which are responsible for firing rate of sino-atrial node. Heart beat in resting position is approximately 70 beat per minute (bpm), it can be lower or higher with respect to physical or emotional condition of the body such as; sleep, illness, exercise, anxiety etc.

In a cardiac cycle, the following operations are carried out;

1. The sino-atrial node fires.
2. Electrical stimulation is diffused along the atrial musculature at relatively low rates, inducing contraction of the atria that can be clearly seen in the P wave given in Figure 2.2. Due to nature of gentle contraction, the P wave is a light, small-amplitude wave of 0.1 - 0.2 mV for 60 - 80 ms.
3. At the (AV) node the excitation signal encounters a diffusion delay that results right after the P wave in the heart signal, known as the PQ fragment. The delay helps to the completion of the blood transfer from the atria to the ventricles.

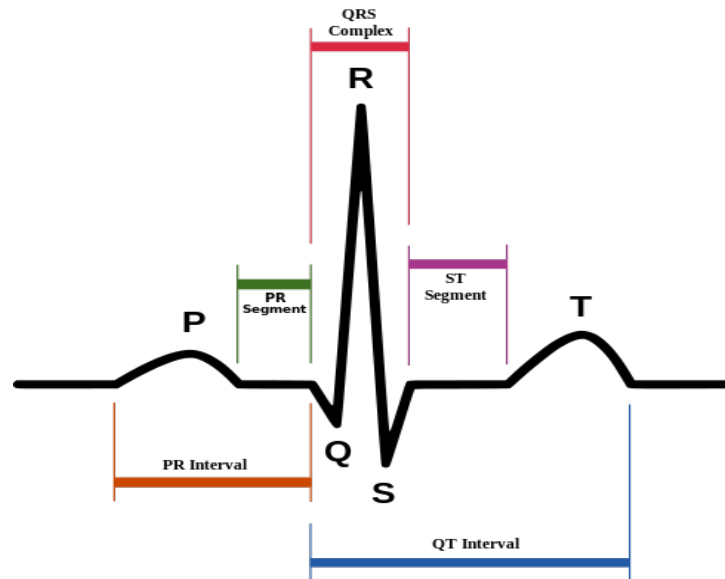


Figure 2.2. P-QRS-T Signal Shape (From Wikipedia 2016)

4. Conduction fibers (The His bundle, the bundle branches, and the Purkinje system) spread and permeate the stimulus to the ventricles at a high rate.
5. The wave of stimulus propagates rapidly from the top of the heart upward, causing quick contraction of the ventricles. This results in heart signal structure as a sharp wave of about 1mV amplitude and 80 ms.
6. Ventricular muscle cells have a comparatively long operation process of 300 - 350 ms as can be seen in Figure 2.2. The plateau part of the operation process leads a normally iso-electric duration of 100 - 120 ms which is ST interval.
7. As ventricles relax, slowly T wave forms which has an amplitude of 0.1 - 0.3 mV for 120 - 160 ms, given in Figure 2.2.

Heart signals are non-stationary signals which include different frequency components.

The common frequency ranges of the important segments of an ECG signal are:

- **HR:** 0.67 to 5 Hz (i.e. 40 to 300 bpm)
- **P-wave:** 0.67 to 5 Hz
- **QRS:** 10 to 50 Hz
- **T-wave:** 1 to 7 Hz
- **High frequency potentials:** 100-500 Hz

and also the common frequencies of the artifact and noise on the ECG signals are:

- **Muscle:** 5 to 50 Hz
- **Respiratory:** 0.12 to 0.5 Hz (e.g. 8 to 30 bpm)
- **External electrical:** 50 or 60 Hz
- **Other electrical:** typically >10 Hz (muscle stimulators, strong magnetic fields, pacemakers with impedance monitoring)

2.2. Electrocardiography

Human body continuously generates signals to regulate its perfect functioning. These signals include information of operations in the body and they can be monitored or stored through instruments like oxygen saturation level detector, blood glucose device, Electrocardiogram (ECG) or Electroencephalogram (EEG). Brain signals or heart signals are the most known ones and real-time monitoring of these signals are carried out with the co-operation of engineers and medical staff (doctors, technicians).

In an ECG instrument, when the sino-atrial node fires, electrical stimulation is propagated as pulsating electrical waves along the heart and the skin as aforementioned in the previous section. These signals are traced via metal disks called electrodes, placed on patient's arms and legs. The electrodes are placed on skin in order to capture the very low impulses of the heart in terms of microvolts. The resulting QRS heart signals can be seen and recorded clearly by ECG devices.

By real time data monitoring and advanced signal processing techniques which can show both time and frequency information together, tracking and diagnosis of heart diseases get easier day by day. Although there are various commercial real-time heart signal monitoring instruments, more user friendly devices should be developed, especially for the elder and children.

In the babyhood for those people who have innate heart diseases, it is even more important to design a safe and simple device without any danger or risk.

2.3. Cardiac Arrhythmias

The term arrhythmia implies that the electrical impulse of the heart is not being propagated along the ventricles of the heart as it should be. The automatic diagnosis and categorization of arrhythmias has been broadly investigated in the last decades in order to facilitate the detection of heart diseases (de Albuquerque et al., 2018).

There are several arrhythmia types accepted in literature. However classification and subtypes of arrhythmias are still being researched yet. According to the Association for the Advancement of Medical Instrumentation (AAMI), heart beats can be categorized in five main groups such as; Normal beat (N), Supraventricular ectopic beat (S), Ventricular ectopic beat (V), Fusion beat (F), Unknown beat (Q). In total, there are 15 subgroups which can be seen in the Table 2.1.

Table 2.1. Heart Beat Classes

AAMI class	Type of beat
Normal (<i>N</i>)	Normal beat Left bundle branch block beat Right bundle branch block beat Atrial escape beat Nodal (junctional) escape beat
Supraventricular ectopic beat (<i>S</i>)	Atrial premature beat Aberrated atrial premature beat Nodal (junctional) premature beat Supraventricular premature beat
Ventricular ectopic beat (<i>V</i>)	Premature ventricular contraction Ventricular escape beat
Fusion beat (<i>F</i>)	Fusion of ventricular and normal beat
Unknown beat (<i>Q</i>)	Paced beat Fusion of paced and normal beat Unclassifiable beat

In this thesis, investigation of Normal beat (N), Left bundle branch block beat (L), Right bundle branch block beat (R) and Paced beat (I) which are defined fundamentally above, is carried out using Discrete Wavelet Transformation (DWT).

2.3.1. Normal

Normal beat (N) refers to the healthy heart beat of an individual which is represented in PQRST waveform, as explained in detail in the previous chapter. Figure 2.3 indicates one second of normal sinus rhythms that are taken from the MIT-BIH arrhythmia database and plotted in MATLAB.

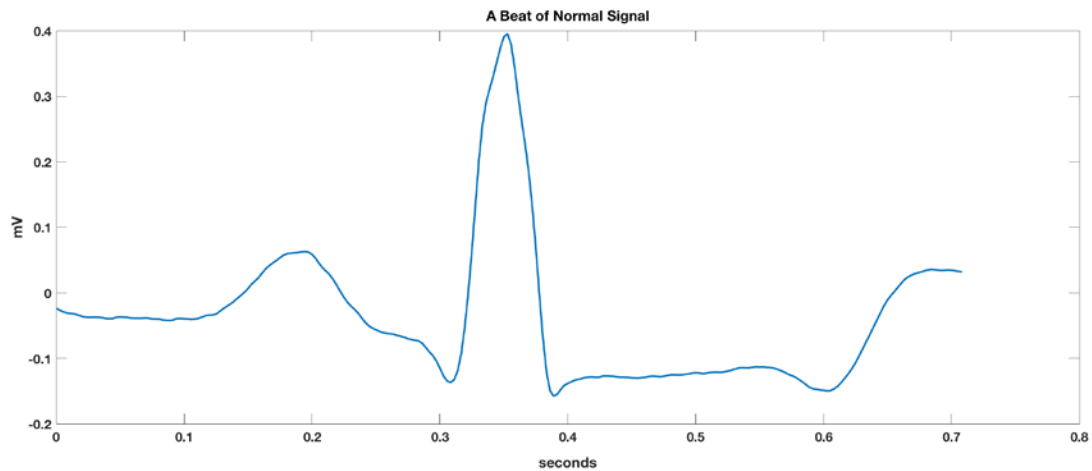


Figure 2.3. Normal Heart Beat

2.3.2. Left Bundle Branch Block

Left bundle branch block (LBBB) is an arrhythmia type which the excitation of the left ventricle (LV) is delayed. It is important because LBBB mostly indicates that a kind of underlying heart disease exists.

LBBB shows characteristic changes on the ECG. Thus doctors can diagnose this abnormality simply by looking at an ECG. A normal QRS complex in a healthy individual, which shows the electrical impulse distributed across the heart, lasts for 0.08 - 0.1 seconds whereas with LBBB it takes usually more than 0.12 seconds. This arrhythmia also appears in QRS pattern as an unexpected upright or downward leads. Figure 2.4 indicates one beat of an LBBB signal that is taken from the MIT-BIH arrhythmia database and plotted in MATLAB.

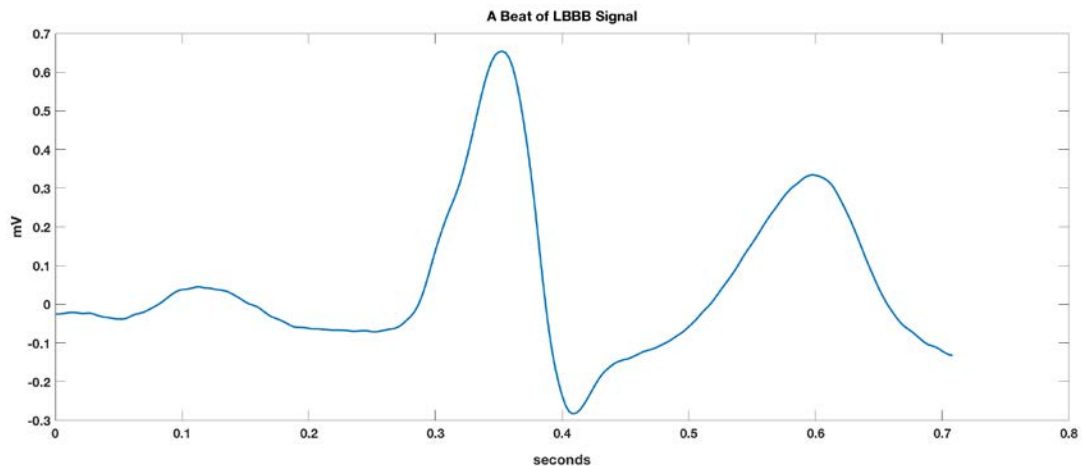


Figure 2.4. Heart Beat of LBBB Rhythm

2.3.3. Right Bundle Branch Block

Right bundle branch block (RBBB) is another arrhythmia type in which a partial or complete blockage of the electrical signal transmission to the right ventricle (RV) occurs. RBBB delays the electrical stimulation of the RV. As a consequence, the right ventricle is activated right after the left ventricle is stimulated so that its contraction is also delayed.

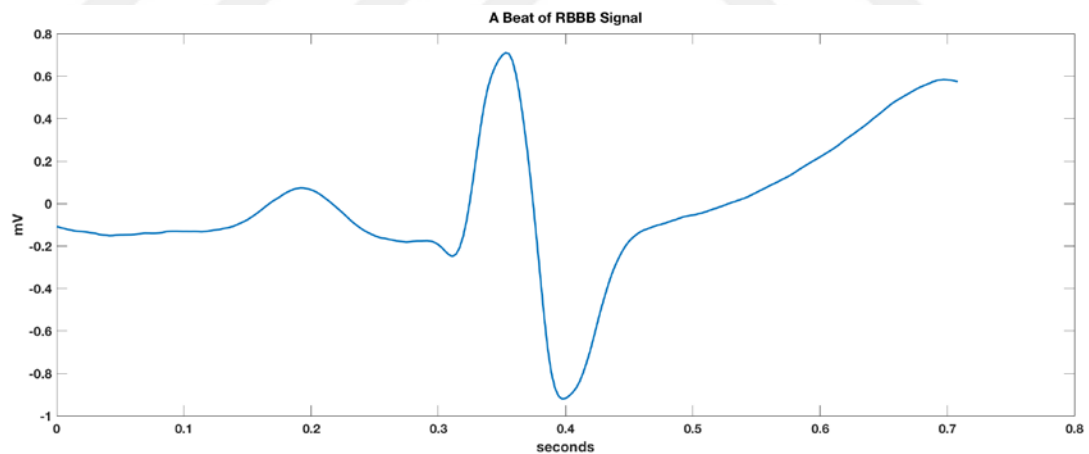


Figure 2.5. Heart Beat of RBBB Rhythm

The QRS complex is wider than normal since the contractions are delayed, therefore it takes longer time than normal for the delayed impulse to be propagated. RBBB is rarely seen in the young whereas more than 11 % of 80 years old individuals are diagnosed with it. In contrast with LBBB, RBBB is diagnosed more frequently but is less important medically. Figure 2.5 indicates one beat of an RBBB signal that is taken from the MIT-BIH arrhythmia database and plotted in MATLAB.

2.3.4. Paced

Heart signals recorded from the people who have an artificial battery in their heart instead of a natural master pace are categorized as paced beat. Figure 2.6 indicates 3 seconds of paced beats which are drawn in MATLAB over and over (Note: this signal is taken from MIT-BIH database).

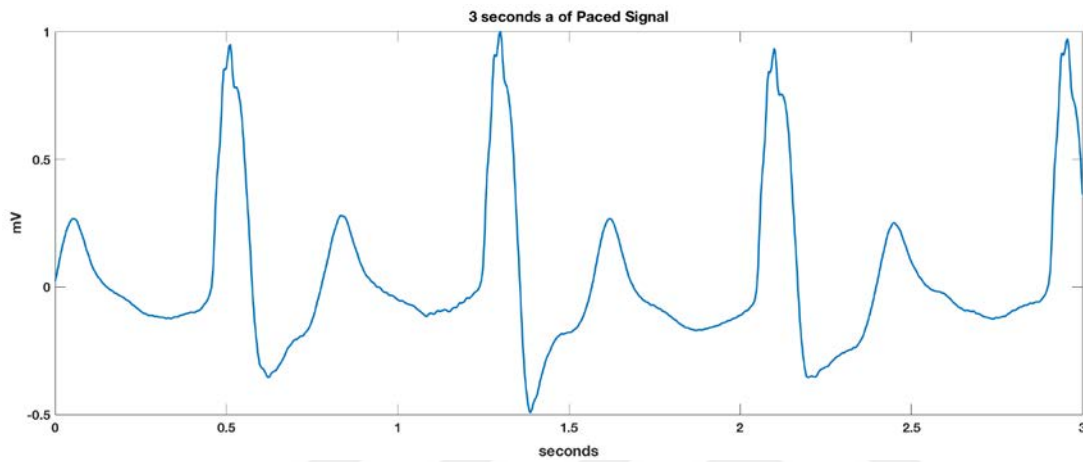


Figure 2.6. Heart Beats of Paced Arrhythmia

CHAPTER 3

SIGNAL PROCESSING AND MACHINE LEARNING

3.1. Wavelet Transform

In daily life, most of the signals we study on are in time domain. Fourier Transform provides the frequency components of that signal. Although it is a strong representation of a signal, the time information is lost. For this reason, finding instantaneous frequency, which is the local frequency at a specific time is important in signal processing. If multi-component non-stationary signals are considered, instantaneous frequency is not suitable enough to understand the character of these kinds of signals. Choosing an analysis window (such as Hamming) and moving it on the signal by applying Fourier transform gives the time-frequency representation with a limited resolution and it is called Short Time Fourier Transform (STFT) (Auger, 1996). However, the STFT falls short where it can not provide both time and frequency resolution at high levels together that you have to choose either one of them; poor frequency resolution with high time resolution or vice-versa.

Wavelet transform (WT) is developed in 1980s as an enhanced method to provide high-resolution frequency and time information which helps to describe signal's characteristics. WT constitutively uses wave functions called wavelets. Structurally expanded or compressed wavelets can also be considered as variable-sized windows on the signal. This is the fundamental difference that distinguishes WT from STFT. Applying WT or namely sliding a variable sized window on a signal gives a detailed representation of the signal. Thanks to the multi-resolution feature of the wavelets, useful information of signals can be investigated. WT can be applied in two ways: Continuous Wavelet Transform (CWT) and Discrete Wavelet Transform (DWT).

3.1.1. Continuous Wavelet Transform

Sliding a scaled wavelet function on a signal along time axis is what basically CWT performs. By scaling, the magnitude of the wavelet is changed and by translation the wavelet is shifted over time axis. If a function satisfies some mathematical criteria,

they can be defined as wavelets, most common examples are; Gaussian Wave, Mexican Hat which is the second derivative of a Gaussian, Haar and Morlet (Addison, 2017).

In mathematical language, convolution of a signal $x(t)$ with a wavelet function which is denoted by $\psi(t)$, gives the wavelet transform of signal $x(t)$. Moving the wavelet function on a continuous signal uses two parameters; translation parameter b and dilation parameter a . CWT is expressed as

$$T(a, b) = \frac{1}{\sqrt{a}} \int_{-\infty}^{\infty} x(t) \psi^* \left(\frac{t-b}{a} \right) dt \quad (1)$$

where ‘*’ denotes complex conjugate of the wavelet function. Translation parameter b gives the location information in time axis and dilation parameter a refers to the scale of the wavelet. Mathematical definition of a wavelet function is written by depending on scale a and location b in equation (1).

$$\psi_{a,b}(t) = \frac{1}{\sqrt{a}} \psi \left(\frac{t-b}{a} \right) \quad (2)$$

The more the wavelet matches the nature of the signal the more detailed information is obtained from the signal. Figure 3.1 shows the analysis of a signal with a wavelet. When the signal $x(t)$ and the wavelet $\psi_{a,b}(t)$ are in opposite regions of time axis like region C,D and E; the integration gives larger negative transform values of $T(a, b)$ given in equation (2) due to low compatibility of the signal and wavelet. On the other hand if the signal and the wavelet are both in the same regions of time axis, the result of $T(a, b)$ gives larger positive transform values due to higher coherence between signal and wavelet like region A and B.

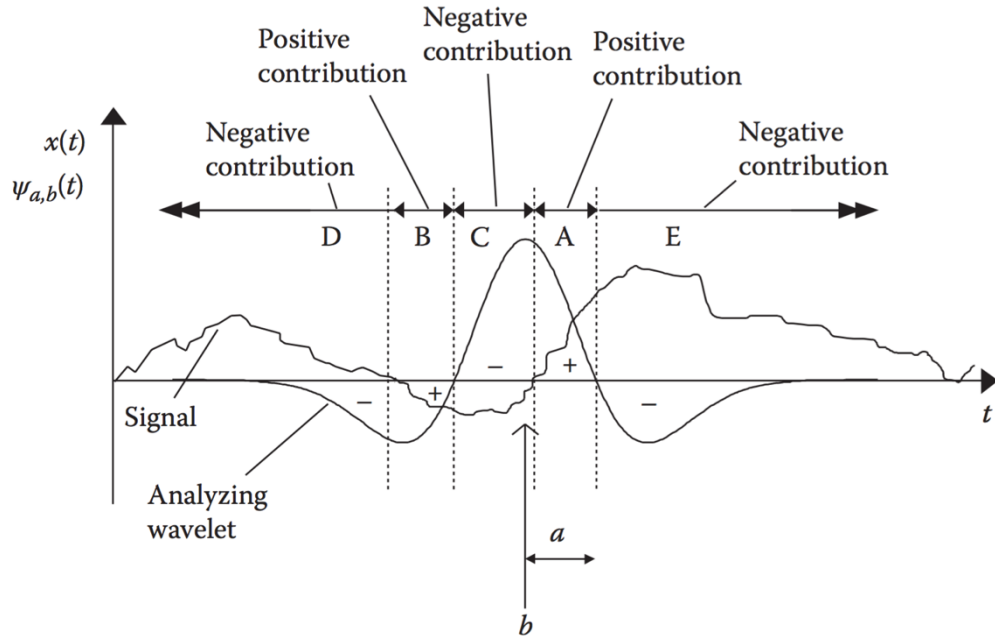


Figure 3.1. Wavelet Analysis of a Signal (From Addison, 2017)

In CWT time, location and scaling parameters can be changed due to the characteristic of the signal. Inverse wavelet transform is performed by integrating over all scales and locations which recovers the original signal from its CWT, as written;

$$x(t) = \frac{1}{C_g} \int_{-\infty}^{\infty} \int_0^{\infty} T(a, b) \psi_{a,b}(t) \frac{da db}{a^2} \quad (3)$$

where $x(t)$ is the original signal and C_g represents admissibility constant which is a variable depending on selected wavelet. Instead of using all values of a , a simple filtering can be performed by restricting the values of dilation parameters a during the process of reconstruction of the original signal. For instance, removing high frequency noises from the signal or decomposition of unified signals with different frequency components.

In order to compute the CWT by integration, the integral in the equation (3) is replaced by a discrete sum which contains the time interval Δt of samples for parameters a and b . Instead of continuous integrals, it is possible to reconstruct the original signal by summing infinite wavelet coefficients. In this context, discrete wavelet transform (DWT) which provides faster computation structure than CWT, will be examined in the next section.

3.1.2. Discrete Wavelet Transform

Discrete Wavelet Transform (DWT) is applied by using orthogonal wavelet basis in discrete steps on a continuous signal. DWT uses discrete value of parameters a and b by moving in each b position with discrete steps, each are proportional to parameter a , which provides connection between a and b . This relation can be described as a wavelet form in equation (4).

$$\psi_{a,b}(t) = \frac{1}{\sqrt{a_0^m}} \psi\left(\frac{t - nb_0 a_0^m}{a_0^m}\right) \quad (4)$$

It is necessary to reflect the fundamental uncertainties of the decomposed data as a result of wavelet transformation in the DWT spectrum. This is done simply by dyadic grid drawing. The data which are obtained from DWT can be expressed in a dyadic grid that contains different size of tiles and these tiles depend on the components of the data's actual time and frequency resolution. By putting the most common values of a_0 and b_0 , which are 2 and 1 respectively, into the equation (5), dyadic wavelet equation can be denoted mathematically as below.

$$\psi_{m,n}(t) = \frac{1}{\sqrt{2^m}} \psi\left(\frac{t - 2^m n}{2^m}\right) \quad (5)$$

where m is the scale index and n is the length of wavelet. The scaling function that makes the signal smoother is absolutely orthogonal to its discrete translation and very similar to a wavelet form, denoted by

$$\phi_{m,n}(t) = \frac{1}{\sqrt{2^m}} \phi\left(\frac{t - n}{2^m}\right) \quad (6)$$

where $\phi_{m,n}(t)$ is the scaling function and is derived from the shift value n in the time axis for m^{th} index of scaling function. The application of the scaling function on to the signal is done by a convolution operation which gives the approximate coefficients below;

$$S_{m,n} = \int_{-\infty}^{\infty} x(t) \phi_{m,n}(t) dt \quad (7)$$

where $S_{m,n}$ represents scaling factor and $x(t)$ is a continuous time domain function.

According to equation (7) in above, if the input signal is finite and limited by certain integers, the equation (7) turns into a specified function given in equation (8).

$$S_{m+1,n} = \frac{1}{\sqrt{2}} \sum_k c_{k-2n} S_{m,k} \quad (8)$$

where c_k represents the scaling coefficients which obtained from equation (6). By multiplying c_k by $1/\sqrt{2}$ gives the high pass filter vector (Addison, 2017). In the same way, by using the approximation coefficients in terms of b_k , detail coefficients can also be calculated,

$$T_{m+1,n} = \frac{1}{\sqrt{2}} \sum_k b_{k-2n} S_{m,k} \quad (9)$$

where b_k is the reconfigured scaling coefficients of c_k . By multiplying b_k by $1/\sqrt{2}$ gives the low pass filter vector.

The discrete input signal is considered to be a scale of level m and the signal length is limited to N . In this context, the scale level is expressed by the input signal length as follows: $N=2^M$. By using a set of finite coefficients that express the wavelet, approximate coefficients of the level $m+1$ are obtained over the previous scale level m coefficients, to initialize the calculation discrete input signal is used such as $S_{0,k} = x_k$ which is the k th index of $x[n]$. A schematic of the discrete signal $x[n]$ with the length of N is given in Figure 3.2.

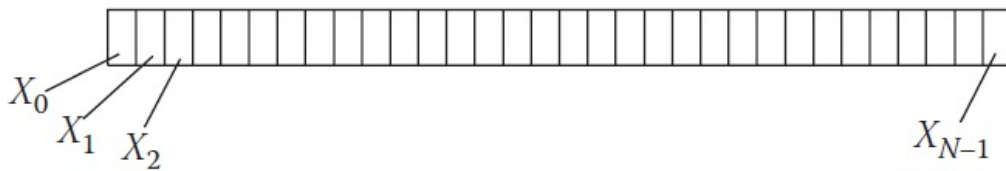


Figure 3.2. A Schematic Representation of Signal $x[n]$ (From Addison, 2017)

By using original discrete signal $x[n]$, DWT computes as Figure 3.3. In the figure below, the original signal is filtered by a low pass and a high pass filter simultaneously. As a result of high pass filter detail coefficients are obtained whereas the approximation coefficients are computed by the low pass filter. For every index scale ,

the approximation coefficients vector is fed into filtering process again and again until it reaches the desired level. Therefore, high resolution signal decomposition has been realized by DWT method.

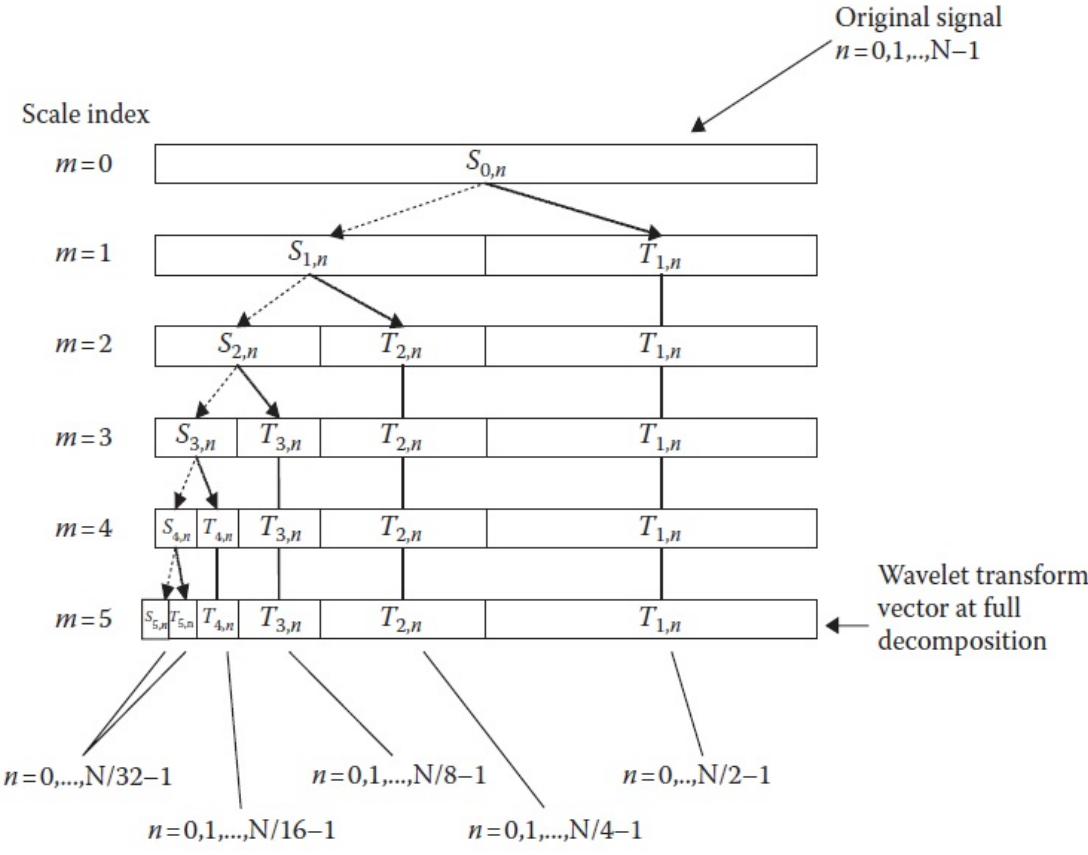


Figure 3.3. Computation of DWT for 5 Scale Index (From Addison, 2017)

The original signal can be obtained by the combination of approximate and detail coefficients.

Algorithm of DWT

The DWT algorithm can be expressed as several versions. The most common version is Mallat’s concept (Mallat, 1989) that described in below steps.

1. Take a signal S with length of N.
2. Selection of a discrete wavelet that suitable for signal S.
3. Use high pass filter and low pass filter coefficients of selected wavelet.
4. Convolve signal S with low pass filter coefficients that obtained from corresponding wavelet. Basically contains a sequences of $(1/\sqrt{2})c_k$ values.

5. Apply step 4 with high pass filter coefficients. Basically contains a sequences of $(1/\sqrt{2})b_k$ values.
6. Result of HPF and LPF apply down sampling by taking $(2n+1)$ th values along the length of the vector.
7. After HPF and down sampling detail coefficients are obtained.
8. After LPF and down sampling approximation coefficients are obtained and the algorithm continues to step 1 by using this step's result.

As a result, signal's atomic decomposition is done by applying filtering process and the Figure 3.4. shows this process schematically.

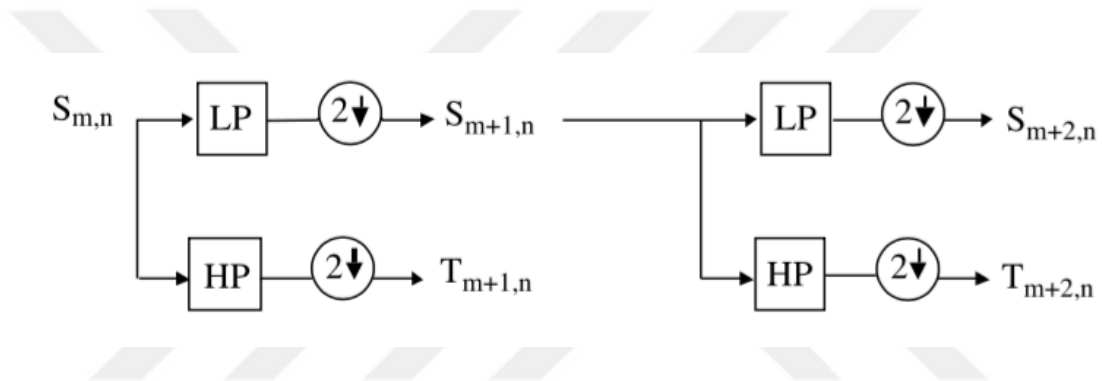


Figure 3.4. Schematic of Filtering Process (From Addison, 2017)

In this thesis, wavelet filter coefficients obtained from Matlab and DWT algorithm was applied both Matlab and Java in same way.

3.2. Neural Networks

Neural Networks (NNs) are inspired by the human nervous system. Nervous system is a mechanism that can be examined in three parts; brain, receptors and effectors. Brain, center of this mechanism, is called neural net. The receptors can forward the stimulus from human body or from the outside environment information, by converting electrical impulses, to the neural net. The effectors allow electrical impulses generated by neural net to convert to perceptible reactions as system outputs. In Figure 3.5. represents basic block diagram of nervous system's working principle (Haykin, 2009).

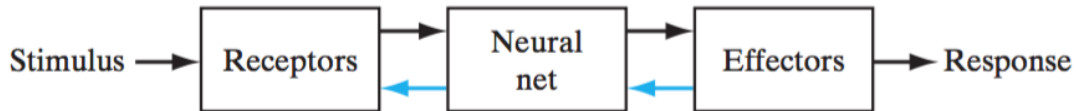


Figure 3.5. Representation of a Nervous System (From Haykin 2009)

Synapse is a connection between nerve cells and other nerve cells. In neural network system synapse is called connection links that shaped by weights and nerve cells is also called as neurons which receive, process and transmit the information in a neural network. Fundamental neuron model contains connecting links, adder and an activation function. In Figure 3.6 shows a nonlinear model of a neuron that include input signals, synaptic weights, bias, adder, activation function and output. Figure 3.6, applied *bias* is expressed by b_k that influences the input of activation function by increasing or decreasing. Neuron is denoted by k and description of the input signals are x_1, x_2, \dots, x_m and synaptic weights are denoted by $w_{k1}, w_{k1}, \dots, w_{km}$ which are the weights of k th neuron, $\psi(\cdot)$ is the activation function; and y_k is the output signal of the k th neuron.

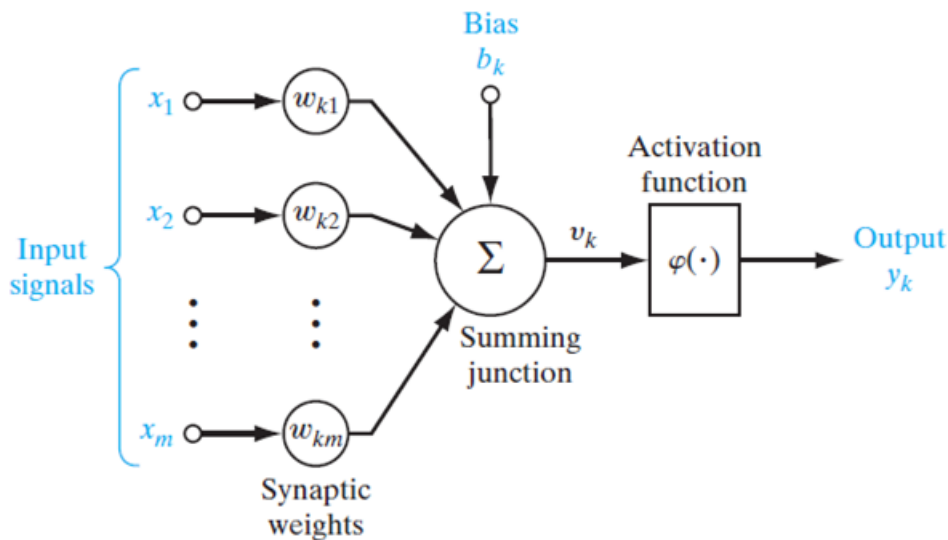


Figure 3.6. Diagram of Non-linear Neuron (From Haykin 2009)

Neurons in a neural network architecture is deeply connected with the learning algorithm which used in training of the network. Learning algorithm is a function that performs the learning process which modify synaptic weights in order to reach

requested design target. Modifying synaptic weights is a general method for neural network designs (Haykin, 2009).

Fundamental Neural Network Structure can be analyzed in three sections; Single-Layer neural networks, Multi-layer neural networks and Recurrent neural networks.

Single-Layer neural networks are the simplest structure that basically contains input layer and output layers as it is shown in Figure 3.7.(a).

In Multi-layer neural networks, there is a layer which contains hidden neurons. It means input and output layer can not see each other directly due to the layer of hidden neurons. Multi-layer networks can contain one or more than one hidden layers. In Figure 3.7.(b) shows a graph representation that is the structure of a multi-layer neural network in the case of a single hidden layer. This graph includes 10 source nodes in input layer, 4 *hidden* neurons in the hidden layer and 2 neurons in the output layer side that shows fully connected neural network.

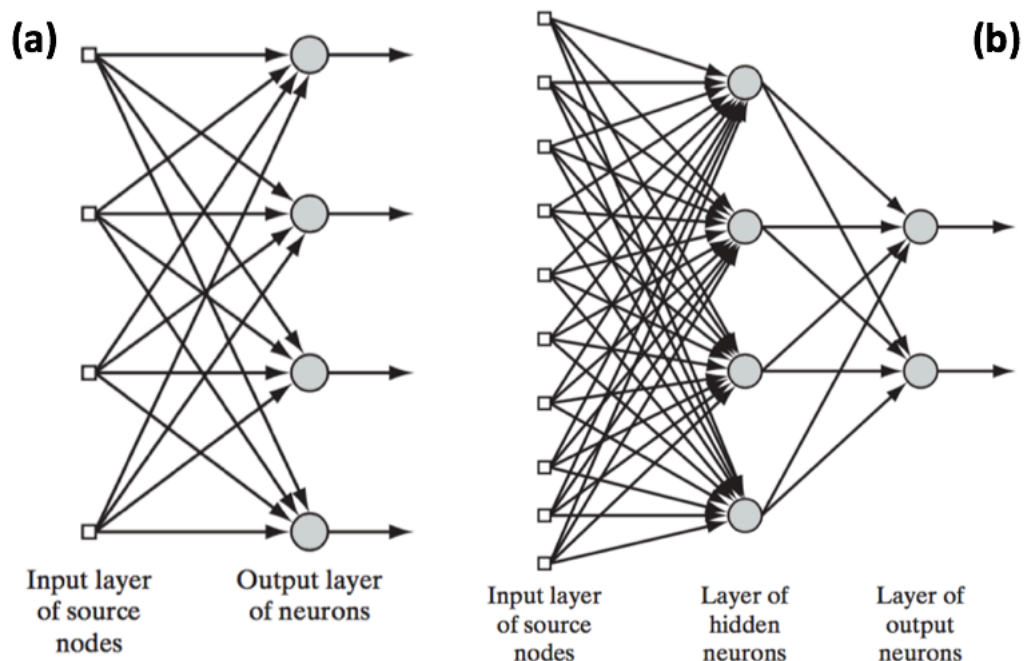


Figure 3.7. Neural Network Models (From Haykin 2009) (a) Graph Representation of Single Layer Neural Network, (b) Graph Representation of Multi Layer Neural Network

Recurrent neural networks include minimum one feedback loop which discriminates it from neural networks. Figure 3.8 shows a diagram of the recurrent neural network.

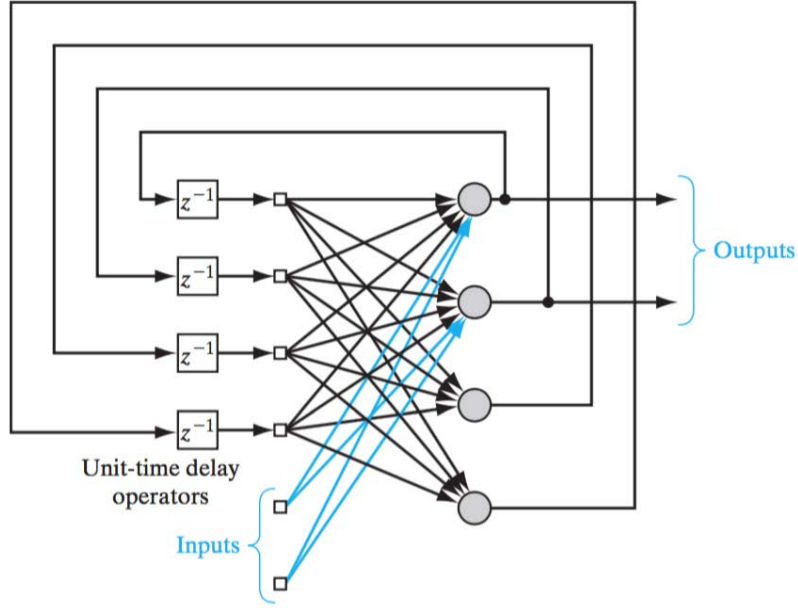


Figure 3.8. Recurrent Neural Network Representation (From Haykin 2009)

The training process of the NN can be realized by three approaches; supervised learning, unsupervised learning and reinforcement learning. That means instances can be labeled or unlabeled. If information about the object is known, neural network will be trained with supervised learning process. Supervised learning has desired output for the input instances. Supervised learning problem can be described by following mathematical notation:

$$\mathcal{T} = \{(\mathbf{x}_i, \mathbf{d}_i)\}_{i=1}^N \quad (10)$$

where \mathcal{T} is the training instances with the size of N , \mathbf{x}_i denotes the input vector and \mathbf{d}_i denotes the desired output which means set of labeled instances.

The error can be calculated by taking difference between desired output and actual output of the neural network. Network's mean square error can be calculated by taking mean value of the error for N instances. Equation (11) shows the mathematical expression.

$$E_{MSE} = \frac{1}{N} \sum_{i=1}^N (d_i - \mathbf{w}x_i)^2 \quad (11)$$

where multiplication of synaptic weight vector \mathbf{w} of sample $i-1$ and sample x_i is the actual output and d_i is the desired output. Supervised learning updates synaptic weight

by minimizing error of each instances. Error of instances is given as feedback to the neural network and train continues in this way. Test data is given to the system that have not seen before in order to evaluate the performance of the trained network.

In unsupervised learning, we can also call self-organized, contains unlabeled data sets or uses environment responses directly. In this case neural network improves new representations of the given data by generating new classes.

In the reinforcement learning, the environment information is transmitted to the learning algorithm and forwarded to the environment as an action at the output of the learning algorithm, in which learning is carried out in such a way that input and output are in continuous interaction.

A neural network has ability to find approximate solutions for large-scale problems. The network is trained by set of data which contains different examples for same object and learn different views of that object. In real problems it distinguishes objects new data according to the its knowledge.

Characterization of the basic information content of an input data set can be achieved by extracting features that do not change into a new form of input. If these features are used, different instances of the same object will emerge. At this point, the number of features applied to the Network can be reduced to realistic levels and consistency is achieved for all objects related to all known transformations. Thus, this is another ability of neural network which is classification of given data. Figure 3.9 shows a block diagram of this ability.

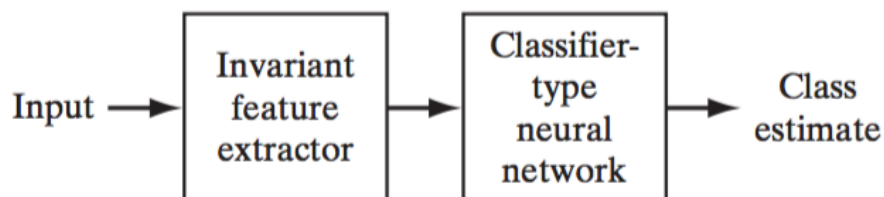


Figure 3.9. Block Diagram Of Neural Network Classification (From Haykin 2009)

There are different types of neural network designs. Perceptron is the basic architecture of a neural network for classification of linearly separable data. If linearly non-separable data is available, different neural network designs must be used such as non-

linear model for classification. The most common type of this model is Multi-Layer Perceptron.

3.2.1. Multi-Layer Perceptron

Multi-layer perceptron (MLP) is a neural network structure, which consist of one or more layers such as Figure 3.10, each neuron in the layer, contains nonlinear activation function, synaptic weights identify the high degree of connectivity.

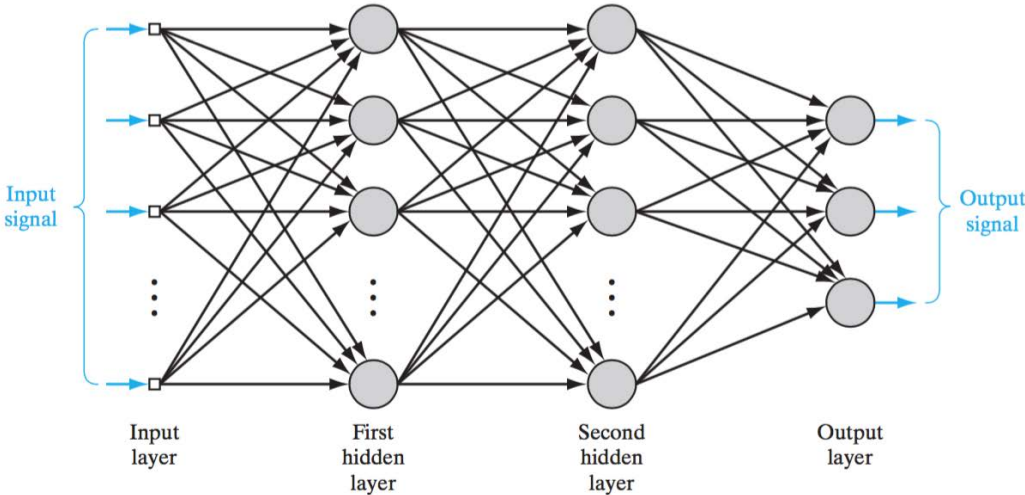


Figure 3.10. Multi Layer Perceptron Scheme with Two Hidden Layers (From Haykin 2009)

Hidden neurons are key factors in the functioning of a MLP. As the learning process progresses, hidden neurons begin to discover step by step through the features that characterize the training data by performing a nonlinear transformation on the input data in order to move towards a new space that's called the feature space.

There are two fundamental methods of supervised learning which are batch and on-line learning. In batch learning synaptic weights are modified epoch to epoch which means average error energy is described as cost function. On the other hand on-line learning is based on training samples. In on-line learning training samples are presented randomly in the network, multi-dimensional space search is performed. It is simply feasible and provides effective solutions for large-scale pattern classifications. Reducing the learning process at the local minimum is an advantage of on-line learning (Haykin, 2009).

Applying the back-propagation algorithm with on-line learning is a method of using for update the weights as sequentially. Mainly the correction of synaptic weights is applied in back-propagation algorithms. Equation (12) shows the mathematical expression of weight correction value.

$$\Delta w_{ij}(n) = -\eta \frac{\partial \varepsilon(n)}{\partial w_{ij}(n)} \quad (12)$$

where change in synaptic weights $\Delta w_{ij}(n)$ can be calculated by taking partial derivative of instantaneous value of the sum of squared errors $\varepsilon(n)$ with weight of synapse j belonging to neuron i , $w_{ij}(n)$ and multiplying it with learning rate η .

MLP can be observed as a sigmoidal structure with back-propagation algorithm.

$$F(\mathbf{x}, \mathbf{w}) = \varphi\left(\sum_k w_{ok} \varphi\left(\sum_j w_{kj} \varphi\left(\dots \varphi\left(\sum_i w_{li} x_i\right)\right)\right)\right) \quad (13)$$

In Equation (13) $\varphi(\cdot)$ represents sigmoid activation function; \mathbf{w} refers to synaptic weights vector which includes the whole set of synaptic weights, w_{ok} refers to synaptic weight from k th neuron of the last hidden layer to the output neuron o . The vector \mathbf{x} represents to input vector and x_i is the i th element of the input vector..

3.2.1.1. Levenberg-Marquardt Method

The design of the multilayer perceptron is specifically to derive the statistical estimation rate with numerical optimization. Several techniques have been developed depending on this. One of them is the Newton's method which's basic principle is to reduce the quadratic approach to the cost function around the weight, which is the key point in every iteration. But due to some difficulties in this method, the quasi-newton method has been developed. The quasi-newton method is not recommended as an efficient method except for small scale artificial neural networks. Another method, the conjugate-gradient method, was introduced to remove the slow speed rate problem in the steepest descent method and also reduce the computational complexity of the Newton method (Haykin, 2009).

The Levenberg-Marquardt method has emerged with the use of structures that converge near the local or global minimum of the Newton method, but also at the same diverge, and slowly converge by ensuring that the correct selection of the step-size parameter of the Gradient descent method. Applying Levenberg-Marquardt method to

a multi-layer perceptron network provides minimizing cost function during the training of the network. The cost function can be defined by

$$\varepsilon_{av}(\mathbf{w}) = \frac{1}{2N} \sum_{i=1}^N [\mathbf{d}(i) - F(\mathbf{x}(i); \mathbf{w})]^2 \quad (14)$$

where \mathbf{d} is the vector of desired output, \mathbf{x} is the input vector, \mathbf{w} is synaptic weights and $F(\mathbf{x}(i); \mathbf{w})$ is the approximating function, returns the weight that corresponds to i th element of \mathbf{x} , is realized by network.

Levenberg–Marquardt method recommends weight modification to the vector of synaptic weights which denoted by \mathbf{w} . Weight modification can be calculated by

$$\Delta \mathbf{w} = [\mathbf{H} + \lambda \mathbf{I}]^{-1} \mathbf{g} \quad (15)$$

where \mathbf{H} refers to Hessian matrix of a second-order function, \mathbf{I} represents to identity matrix which has equal dimension as \mathbf{H} and \mathbf{g} is the gradient vector.

Assume that Levenberg-Marquardt method contains n iterations. It computes $n-1$ iteration's $\varepsilon_{av}(\mathbf{w})$ then chooses a λ value. Calculate the adjustment weight value of $\Delta \mathbf{w}$ at iteration n and recalculate the cost function by adding $\Delta \mathbf{w}$; $\varepsilon_{av}(\mathbf{w} + \Delta \mathbf{w})$. If the value of $\varepsilon_{av}(\mathbf{w} + \Delta \mathbf{w})$ is equal or greater than $\varepsilon_{av}(\mathbf{w})$, increase the λ value and go back to the $\Delta \mathbf{w}$ calculation by not changing $\varepsilon_{av}(\mathbf{w})$. If the value of $\varepsilon_{av}(\mathbf{w} + \Delta \mathbf{w})$ is lower than $\varepsilon_{av}(\mathbf{w})$, decrease the λ value go back to the $\Delta \mathbf{w}$ calculation by updating \mathbf{w} value to $\mathbf{w} + \Delta \mathbf{w}$.

3.3. Evolutionary Algorithms

Evolutionary Algorithms (EAs) are based on the natural selection concept of Darwin. According to Darwinian theory, an individual competes, to survive and reproduce. To understand the process steps of the EAs, it is necessary to define the metaphors associated with EAs. As in Darwinian theory problem is the environment, solution is the individual and objective function is the fitness so the fittest individual of defined environment survives (Talbi, 2009).

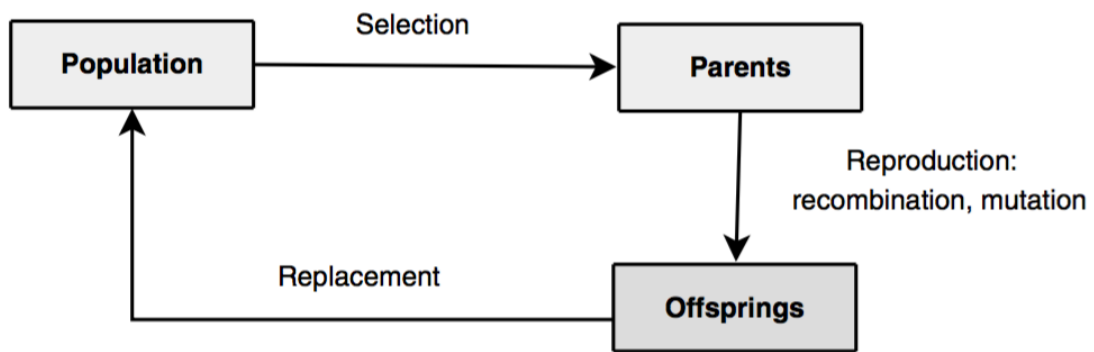


Figure 3.11. Basic Parts of EA (From Talbi 2009)

EAs contains iterative algorithm structure that based on the evolution of species. Common concept of EA has iterations and each iteration consist of a loop that starts with initial population which is usually constituted randomly. Individuals selected from the initial population to constitute the parents and offsprings are reproduced from these parents by using several operators like mutation or recombination. Replacement specifies to which individuals will survive within offsprings and parents. Finally, survivals generate to new population. These iterations continue until the specified termination criterion is met. The termination criteria can be the best individual or the limited number of generations. Algorithm 1 is the model of an evolutionary algorithm.

Algorithm 1. Fundamental Model of Evolutionary Algorithm (From Talbi 2009)

```

Generate( $P(0)$ ) ; /* Initial population */
 $t = 0$  ;
While not Termination_Criterion( $P(t)$ ) Do
  Evaluate( $P(t)$ ) ;
   $P'(t)$  = Selection( $P(t)$ ) ;
   $P'(t)$  = Reproduction( $P'(t)$ ) ; Evaluate( $P'(t)$ ) ;
   $P(t + 1)$  = Replace( $P(t)$ ,  $P'(t)$ ) ;
   $t = t + 1$  ;
End While
Output Best individual or best population found.
  
```

In biology, all gene sequencing is called genotype and a set of alleles that determines the appearance of a particular feature or property is called phenotype. EAs genotype denotes element of the solution and phenotype denotes the solution. When the operators like mutation, recombination are applied to genotype level, the fitness function uses the phenotype of the associated individual (Figure 3.13).

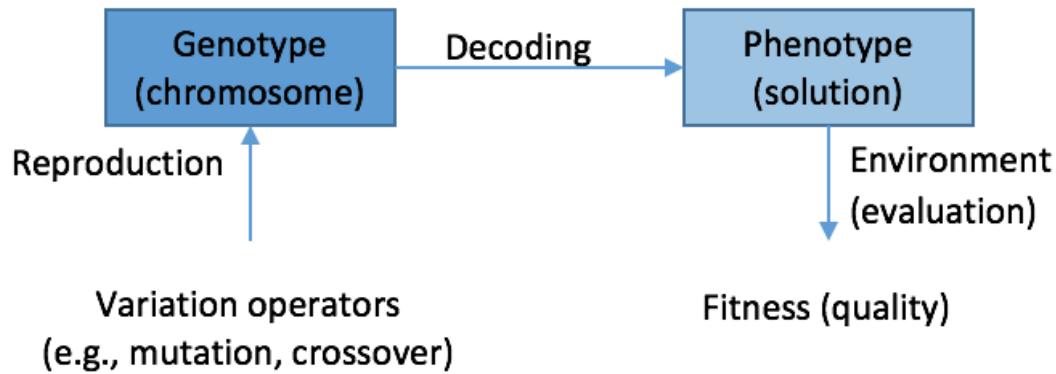


Figure 3.12. Block Diagram of Phenotype Corresponding to the Genotype (From Talbi 2009)

3.3.1. Genetic Algorithm

Genetic Algorithms (GA) is a subclass of evolutionary algorithms. GA is the method of searching for the most optimal or acceptable solution of a problem.

GA contains specialized structure of EA. Individuals are represented as different kind of data structures which are related to each other and selected from population depending on the fitness rate. Mutation operator is applied to change the content of the individual randomly with the aim of increasing diversity among the solutions. N-point or uniform crossover is used to generate new offspring and all offsprings are replaced by parents (Talbi, 2009).

Concepts of Genetic Algorithms

A) Representation

Generally, in GAs encoded solution denotes a chromosome and decision variables in the solution is called genes. Each decision variable impacts chromosome and chromosome changes the individual. There are three type representation in EA concept which are discrete representation, order-based representation and binary representation. Binary representation is usually used in GAs. Binary system values are used as genes and the combination of 1 and 0s provide chromosome.

B) Population

Population represents the subset of the possible solutions. Initial population can be created in two way: randomly generated solutions or generated from the problem's known solution.

C) Fitness Function

Fitness function in GA, takes a candidate solution of defined problem as input then produces an output that shows how solution fits the problem and how good a solution is for that problem. Calculation of the fitness value is done for each individual consecutively therefore calculation should be done fast. To produce fittest solution of the problem needs to quantitatively measure how individual can be produced from given solutions.

D) Selection Methods

In GAs there are many different methods used in selecting the individuals to be copied in the next generation. Main idea of the selection is the better individual has the more chance to be a parent. This kind of selection tendency will lead the population to find better individuals. On the other hand the worst individuals of the populations should not be ignored against the possibility of creating a better individual in future generations.

Selection is a mechanism that decides which individual is used for reproduction and how many offspring the associated individual can be produce. Rank-based selection, roulette-wheel selection, tournament selection and stochastic universal sampling are some of these methods. Combinations of these methods can be used also.

Rank-based selection is guaranteed to select the most suitable individuals from each generation, each generation's best individual or a few best individuals is copied to the next generation for the possibility that a better individual may not emerge when the termination criteria is reached.

Roulette-wheel selection, individuals related with the fitness value, in which the suitability of the new generation to be selected is determined on the basis that the individual has higher or lower fitness value than its competitors.

Stochastic Universal Sampling, has similar description in roulette-wheel selection however, it distinguished from roulette-wheel selection in that it makes more than one selection and each selected individual has equal distances to each other.

Tournament selection, the subgroups of the individuals are formed from the larger populations, and all the subgroups compete with each other. Only one individual from each subgroup is selected as a candidate for crossover to form a new population.

E) Crossover Methods

Common concept of crossover are single-point, n-point, and uniform crossovers. In the one-point crossover, as shown in Figure 3.14, the k value which is the crossing point, is randomly selected and two roots are created by changing the cuts of the parents and two new individuals are created by exchanging these parts of the parents. In the same way, the generation of two new individuals by randomly determining multiple k points is called n-point crossover.

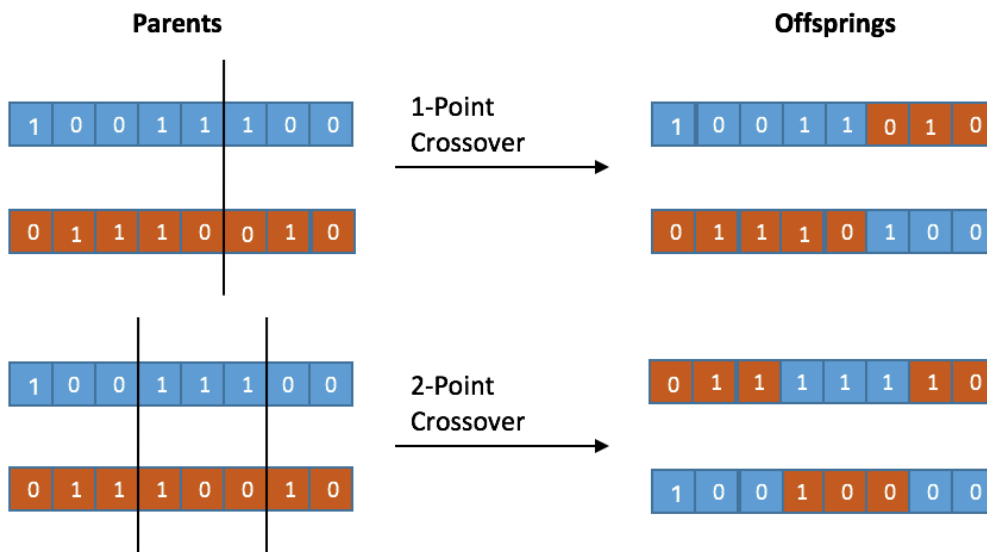


Figure 3.13. One point and Two Point Crossover Representation

Two parents using one-sample crossover can be recombined according to the size of their chromosomes. Each element of the chromosome is randomly selected. As indicated in Figure 3.15, each parent equally contributes to produce new individuals.

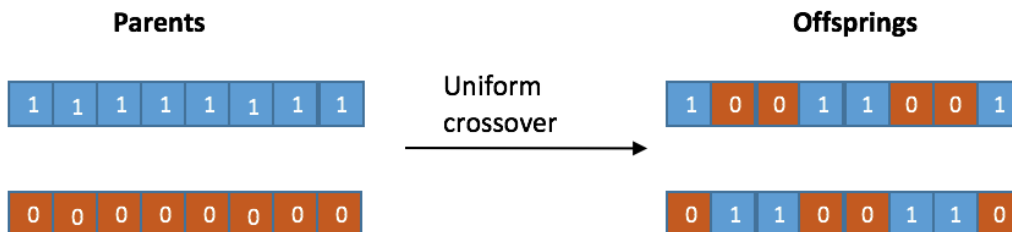


Figure 3.14. Uniform Crossover Representation

F) Mutation Methods

After selection of individuals using the selection method, random changes are

implemented in the hope of increasing the suitability of the individual to be transferred to the new individual. The simplest of these is called mutation. Just as mutations in living things change a gene, in a genetic algorithm, mutation is expressed as a small change in the single points of the individual's code.

Frequently used mutation methods are bit flip, random reset, swap mutation. Random reset is performed by assigning a random value to a group of randomly selected genes. In the swap mutation, we randomly select two positions on the chromosome and change their values. As bit flip operation that mostly used for binary representation. This operation changes the genes value to opposite.

3.3.2. Multi-Objective Genetic Algorithm

In standard GAs, the fitness function is determined by looking at a single criterion. However, a single criterion for some data may not provide much information and more than one criterion should be assessed. At this point, a multi-objective optimization strategy becomes necessary. The multi-objective approach proposes to look at multiple criteria in the search for appropriate solutions (Fonseca and Fleming, 1993). In this context, the fundamental difference between GA and Multi-objective optimization is the assignment and evaluation of the fitness function.

Multi-objective approach to GA search more than one set of candidate solutions searching for the best solution. In fitness function assignment consists of four types of multi-objective optimization approaches; scalar, dominance-based, indicator-based, and criterion-based.

Scalar Approach: It is based on the conversion of the multi-objective problem into a single-objective problem. This approach is a collection-based method that combines multiple targets into a single function.

Dominance-based Approach: The search uses the notion of dominance and Pareto optimality to lead the process. The objective vectors of the solutions are scaled using the dominance relation.

Indicator-based Approach: Uses performance quality indicators to direct the search towards the Pareto front.

Criterion-based Approach: The search is carried out by taking several non-measurable targets separately.

In this thesis, a scalar approach was used to assignment of the fitness function in GA with multi-objective approach.



CHAPTER 4

ARRHYTHMIA DETECTION

4.1. Data Set

The MIT-BIH Arrhythmia Database was first established as a standard test material for assessing arrhythmia findings and has been used for basic research on heart dynamics at approximately 500 locations in worldwide since 1980.

Long-term ECG recordings obtained by Boston's Beth Israel Hospital Arrhythmia Laboratory are of great importance in the control and development of the accuracy of methods using many studies (Moody and Mark, 2001).

In this thesis, ECG data used for training and testing, was taken from the MIT-BIH Arrhythmia database that was downloaded from Physionet website (MIT-BIH Arrhythmia Database Directory, 2016). This database directory contains 134 records sampled at 360Hz. These records routinely consist of clinical waveforms and some complex records.

134 records of MIT-BIH Arrhythmia database directory labelled with numbers between 100 and 234. In this study 16 of them was chosen from the database and Table 4.1 indicates the detail information such as gender and age of the person. Also the table contains the beat type, heart rate which was measured by using 3 R-R intervals.

Each record of chosen 16 is converted to a “.mat” file as 30 seconds durations for usage of MATLAB. 50 file were created from one record. 200 of them is Normal, 200 is LBBB, 200 is RBBB and the 200 of them are Paced records.

Table 4.1. MIT-BIH Records with Details

Record No	Gender	AGE	Beat	Heart Rate (bpm)
100	Male	69	Normal	70-89
101	Female	75	Normal	55-79
105	Female	73	Normal	78-102
112	Male	54	Normal	74-91
109	Male	64	LBBB	77-101
111	Female	47	LBBB	64-82
207	Female	89	LBBB	57-90
214	Male	53	LBBB	49-92
102	Female	84	PACED	68-78
104	Female	66	PACED	70-78
107	Male	63	PACED	68-82
217	Male	65	PACED	65-76
118	Male	69	RBBB	54-91
124	Male	77	RBBB	47-64
212	Female	32	RBBB	63-108
231	Female	72	RBBB	49-69

4.2. Pan Tompkins Algorithm for QRS Detection

Biomedical signals give information about physiological events. These events are important to identify the diseases. To detect these events, the structure of the signal must be examined and segmented. There is more than one method to examine different type of biomedical signals. Pan Tompkins algorithm is one of them which is proposed for heart signals. The main principle of this algorithm is to detect QRS complexes from a real-time ECG signal (Pan and Tompkins,1985). In Figure 4.1 is a block diagram representation of Pan Tompkins algorithm which includes bandpass filter, derivative, squaring function and moving-window integrator.

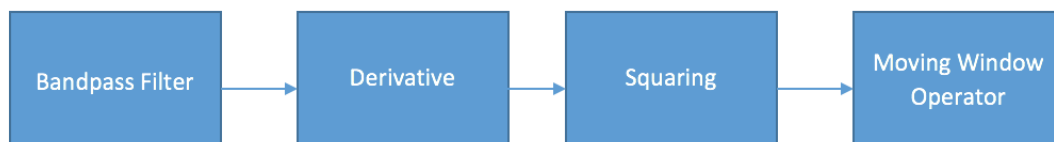


Figure 4.1. Block Diagram of Pan Tompkins Algorithm

Pre-filtering is performed on the signal that has been recorded for the detection of the QRS complex in the content of the ECG signal. Power-lines generate interferences during transmission over signals having values in the order of mV, such as an ECG

signal. The low pass filter (LPF) successfully removes noise generated by the power line from the signal through the ECG cables. The transfer function of the LPF applied in software, is expressed in the following equation (Rangayyan, 2015).

$$H_{lp}(z) = \frac{(1 - z^{-6})^2}{(1 - z^{-1})^2} \quad (16)$$

The filter was created both Matlab and Java algorithms with the transfer function which is denoted in equation (16). Applying the LPF to the signal as a digital filter is mathematically defined as the convolution of the transfer function on a signal. The convolution can be denoted as

$$y(n) = \sum_j x(j)h(n - j + 1) \quad (17)$$

where $y(n)$ is the output and vector h (array of filter coefficients) shifts across the vector x (the signal vector) which refers to overlapping area of the points. The filter allows components with 60 Hz to be removed from the signal.

To complete the bandpass filtering on a signal, high pass filter (HPF) applied after a LPF. The transfer function of the HPF, applied in software, is specified in the following equation.

$$H_{hp}(z) = z^{-16} - \frac{(1 - z^{-32})}{(1 - z^{-1})} \quad (18)$$

The convolution process which is expressed in equation (17), applied in the LPF, is carried out in the HPF. Cut-off frequency of the HPF is 5Hz and it constitutes a delay on a signal.

Getting derivative of the filtered signal is the next step of the Pan Tompkins algorithm. The derivative of a signal vector can be calculated by using following function

$$y(n) = \frac{1}{8} [2x(n) + x(n - 1) - x(n - 3) - 2x(n - 2)] \quad (19)$$

Computing $y(n)$ output signal is basically convolution of the input signal of $x(n)$ with an impulse response.

Squaring function moves all data points to the positive region and makes nonlinear amplification of the output in order to indicate high frequency components, which are results of derivative operation. Squaring is applied each component of input signal as

$$y(n) = x_i^2 \quad (20)$$

where x is the input vector (obtained array after the pre-filtering) and equation (20) computed for $i = 0,1,2 \dots N$ and $n = 0,1,2 \dots N$.

In moving-window integration is a kind of filtering algorithm that removes unexpected peaks that are the result of the derivative operation. Fundamentally, it is a purifying process by sliding a window on the output vector of squaring function. The windowing filter can be written as

$$y(n) = \frac{1}{N} [x(n - (N - 1)) + x(n - (N - 2)) + \dots + x(n)] \quad (21)$$

N refers to the window width. If the window width is chosen a long size, the output result will be give mixed QRS and T waves. On the other hand if it is selected too small value, result of output could contain multiple peaks for each QRS. Suitable value of the width has been found as 30 for the frequency value of 200Hz (Rangayyan, 2015).

Pan Tompkins algorithm includes two types of threshold; to detect a peak as noise and QRS. After window integration, the R-R average value is calculated and multiplication with the maximum value of window integration result gives the low and high limit of the R-R interval. R-R interval limits is searched in the output $y(n)$ that denoted in equation (21). The maximum value within these limit ranges is searched within the input signal and the point at which it is located, is determined as the R peak. Thus, this provides a dynamic structure that threshold is found to detect R peaks by applying an adaptive method to algorithm. This is also called as adaptive thresholding.

In this thesis Pan Tompkins algorithm applied to the each ECG signal for detect QRS complexes and has been implemented both Matlab and Java code.

1. The ECG raw data have been imported from MIT-BIH database to Matlab by converting '.mat' file. Same process has been done for java by converting '.mat' files to '.txt' files. Figure 4.2.(a) indicates the raw data of normal ECG signal rhythm of the MIT-BIH database as Matlab result.
2. DC components have been cancelled out by calculating the mean value of input ECG signal and subtracting this mean value from each component of the ECG signal.
3. Absolute maximum value of the signal has been calculated and each component of the signal has been divided into this value in order to normalize

the signal to the scale of one. Figure 4.2.(b) shows the Matlab result of step 2 and 3.

4. Applying convolution with LPF coefficients vector and the input signal. Normalize the signal for convenience. The Matlab result of output signal is shown in Figure 4.2.(c).
5. Repeat the step 4 for HPF coefficients vector. The Matlab result of output signal is shown in Figure 4.3.(a).
6. For derivation of the signal, create an impulse response coefficients vector and repeat the step 4 by using these coefficients. The Matlab result of output signal is shown in Figure 4.3.(b).
7. Take the square of each value of the input signal and repeat the step 3 (Figure 4.3.(c)).
8. Repeating step 4 by using window filter coefficients vector in order to apply the window integration.
9. After these 8 steps, finding peak level by using procedure of adaptive thresholding.
10. Locations of the R peaks are detected by using peak levels. Extraction of the QRS complex can be completed by taking 127 sample before and 128 sample after the R peak. As a result, QRS complexes, each of contains 256 samples, are obtained from an ECG signal with 360Hz frequency. Figure 4.4 illustrates QRS locations of the ECG signal (Figure 4.4.(a)) that founded by Pan Tompkins algorithm and single QRS complexes that extracted from 30 seconds ECG signal (Figure 4.4.(c)).

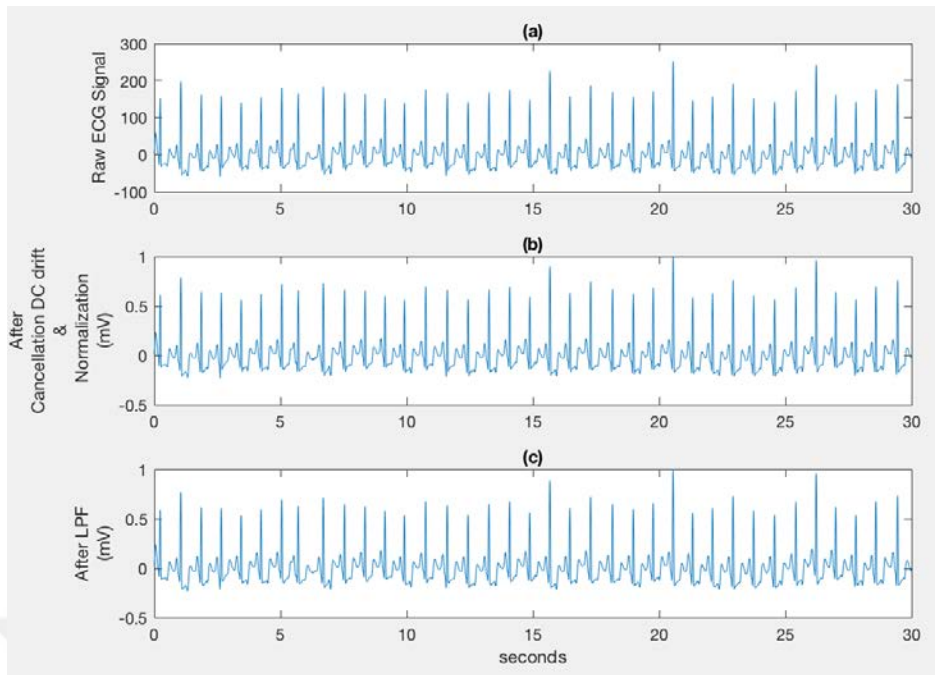


Figure 4.2. Application of Pan Tompkins Algorithm on ECG Signal (a) Raw Data of ECG Signal, (b) Cancellation DC drift and Normalization of Raw ECG signal which normalized to one,(c) Signal (b) After Applying LPF

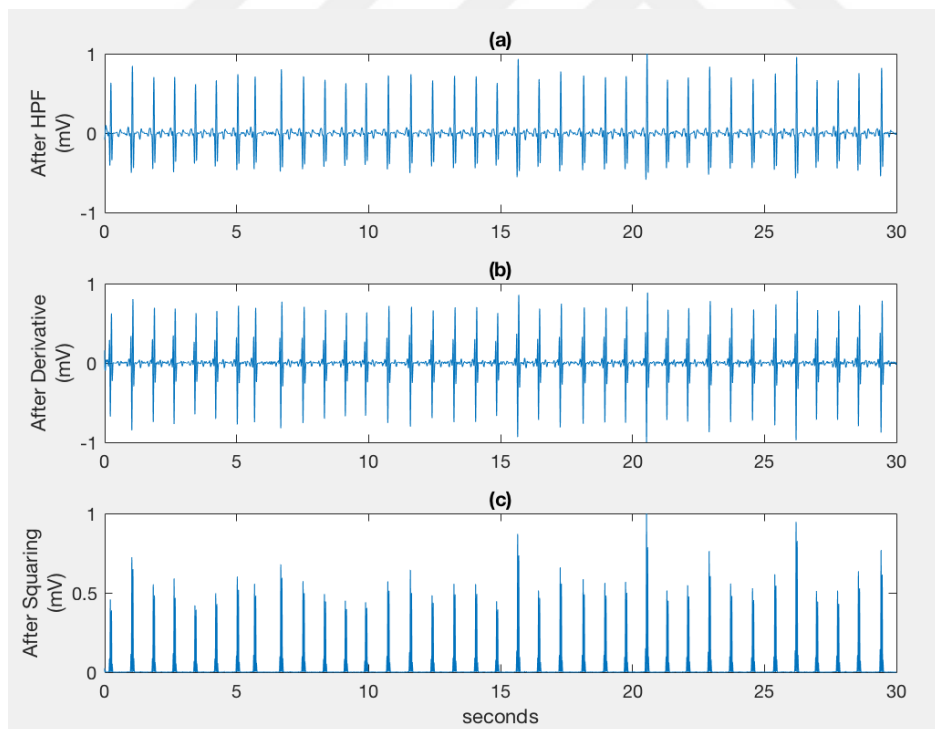
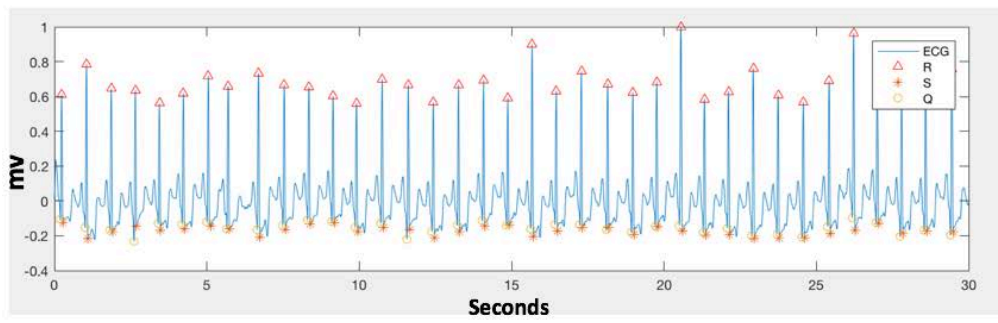
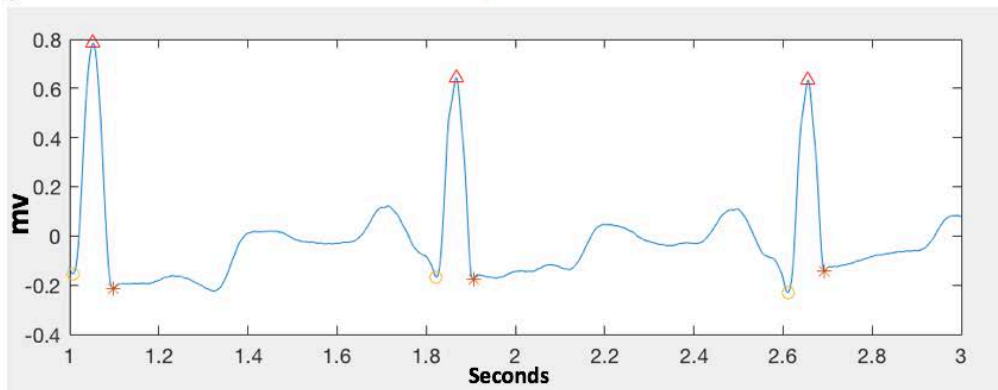


Figure 4.3. Finding R-peaks of an ECG Signal; (a) After HPF Process of Figure 4.2(c), (b) Taking Derivative of (b), (c) Squaring Result of (b)

(a)



(b)



(c)

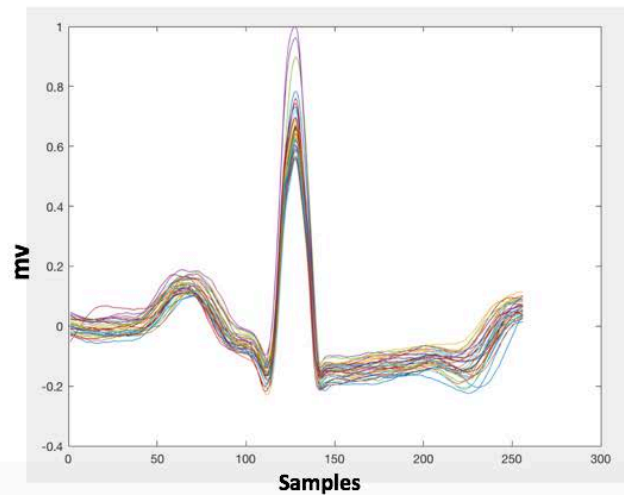


Figure 4.4. Detection and Extraction of QRS; (a) Q-R-S locations in 30 seconds ECG Signal. (b) 3 seconds interval of (a). (c) Single QRS complexes.

Algorithm of QRS Detection: The algorithm detects the QRS blocks of an ECG signal which named by vector \mathbf{x} . In algorithm the length of \mathbf{x} vector is denoted by N . The range in which the R peaks are located is expressed as the array named by pos where the start and end values are stored. M is referred the length of pos array. $RLoc$ represents the array where the locations of the R peak located on the input signal \mathbf{x} . To avoid losing the main signal data of \mathbf{x} , \mathbf{y} vector is the array to which the Pan Tompkins algorithm changes are assigned. The algorithm includes three functions mean, maximum and convolution. The function $mean(\mathbf{a})$ takes an array \mathbf{a} and returns average value of that array contents. The other function maximum is denoted $max(|\mathbf{a}|)$ and finds the maximum absolute value of an element of the array \mathbf{a} then return it. Convolution is a function that takes two vectors and denoted by $conv(\mathbf{a}, \mathbf{b})$. The second vector \mathbf{b} written in the function is shifted along the first vector \mathbf{a} . The convolution result of the two vectors is returned. In this algorithm, new values are assigned to \mathbf{y} vector by taking convolution with low pass filter coefficients \mathbf{u}_{lp} , high pass filter coefficients \mathbf{u}_{hp} , impulse response coefficients \mathbf{u}_{ir} , respectively, of input signal \mathbf{x} . The vector \mathbf{u}_{wf} represents filter coefficients of a window. The variables $thresh$ and $maxh$ is used for hold mean and maximum value of the vector \mathbf{y} after Pan Tompkins algorithm main steps. The $temp$ and $found$ variables are temporary variables that created for loop and condition checks. The variable i, j and k are the subscripts of the arrays.

Algorithm 2. QRS Detection

```
1. [Initialize the data]
2. [Cancellation of DC components]
   i ← 0
   while i < N
       x[ i ] ← x[ i ] - mean(x)
3. [Normalization]
   i ← 0
   while i < N
       x[ i ] ← x[ i ] / max(|x|)
4. [Convolution]
   y ← conv( x, utp )
   perform step 3
   y ← conv( y, uhp )
   perform step 3
   y ← conv( y, uir )
   perform step 3
5. [Squaring]
   i ← 0
   while i < N
       y[ i ] ← y[ i ] * y[ i ]
6. [Moving Window]
   y ← conv( y, uwf )
7. [Adaptive threshold]
   thresh ← mean(y)
   maxh ← max(|y|)
8. [Find R Positions]
   i ← 0
   k ← 0
   found ← 0
   temp ← 0
   while i < N
       if (y[ i ] > thresh * maxh )
           then found ← 1
           else found ← 0
       if (temp != found )
           then if ( found == 1 )
               then pos[ k ] ← i
                   k ← k + 1
               else pos[ k ] ← i - 1
                   k ← k + 1
           temp ← found
9. [Find R locations]
   i ← 0
   k ← 0
   j ← 0
   while i < M/2
       found ← x[pos[ i*2 ]]
       k ← pos [ i*2 ]
       while k < pos[ (i*2)+1 ]
           if (x[ k ] > found )
               then found ← x[ k ]
           end if
       end while
       RLoc[ j ] ← found
       j ← j+1
```

4.3. Features and Feature Extraction

In our world people recognize the materials by physical properties such as colors or shapes. According to their natures, people can identify the physical structures around them through their five sense organs. On the other side in computer world, artificial systems are designed by copying the natural sense of people. In this point, it is necessary to transfer the properties as data which belongs to environment due to distinguish structures by using an artificial system. These properties called as features.

Signals have their own structures and we can differ the signals by looking at their shapes. For instance, an engineer examines the circuit signal on the oscilloscope and finds the problem or a doctor analyses a heart signal on an ECG device and diagnoses the disease. In this context, designation of the signal characteristic, the features must be specified for artificial systems.

Non-stationary signals include different frequency components. For this reason, it is observed that the statistical values of the signal change over time such as mean, variance or higher-order statistics. Taking a non-stationary signal and investigate these values can be meaningful. Previous chapters were mentioned that several frequency components can be observed at different time intervals in a non-stationary signal and the technique which is giving them with high time-frequency resolution is DWT.

ECG signal is a non-stationary signal in terms of including different frequency ranges (was explained in chapter 2). For analyzing an ECG signal, the features which describe the signal well must be specified. The most common features that used for an ECG signal are statistical and qualification values. However, if these features extracted from a signal by using DWT technique, it provides high distinctiveness.

Calculating mean or average value of a signal indicates the magnitude of a particular time interval. After DWT of a signal, detail coefficients give high-resolution frequency component in a time interval. The mean value of l th level detail coefficients can be calculated as

$$M = \frac{1}{N} \sum_{n=0}^N cD_{l,i} \quad (22)$$

where N is the length of the detail coefficient vector and $cD_{l,i}$ refers to i th data of l th level detail coefficients vector. Summation of all data points and dividing this value to

number of data points give the mean value M . Standard-deviation gives us an another statistical information about the distribution of data relative to the average value of the data set. Standard-deviation calculated by taking square root of variance that represents how data in the set are distributed around the mean value. Applying standard-deviation to detail coefficients can be written as

$$\sigma = \sqrt{\frac{1}{N-1} \sum_{i=1}^{N-1} (cD_{l,i} - M)^2} \quad (23)$$

where σ denotes the standard-deviation, M is the mean value and in this case $cD_{l,i}$ represents the data point of l th level detail coefficient of the signal. Entropy is the other statistical feature of the signal that gives the randomness. In this thesis entropy is used to express the impairment and uncertainty of the non-stationary signal. In this context, the probability distribution of the signal is used as a criterion in calculating the entropy of the signal and is applied to detail coefficients of the signal as follows.

$$E(\mathbf{cD}_l) = - \sum_{i=0}^N p(cD_{l,i}) \log_2(p(cD_{l,i})) \quad (24)$$

where $cD_{l,i}$ refers to element of the \mathbf{cD}_l vector (level l detail coefficients) and $p(cD_{l,i})$ is the probability function that counts how many of the same values are in the vector \mathbf{cD}_l .

Energy of a signal can be calculated by summing power of each sample of that signal. To obtain better information about the signal, energy is calculated in wavelet domain by using detail coefficients. In equation (25) energy E can be calculated by taking square value of each detail coefficients that corresponds to l th level $cD_{l,i}$ and summing them.

$$E = \sum_{i=0}^N (cD_{l,i})^2 \quad (25)$$

This thesis proposes to use the features of mean, standard-deviation, energy and entropy which are extracted of DWT signal.

QRS complexes obtained from the Pan Tompkins Algorithm are transformed with

DWT. Daubchies (DB), Symlet (SYM), Coiflet (COIF) and Bio-orthogonal (BIOR) mother wavelets were used to constitute the features. Totally 208 features were constructed as mean, standard-deviation, energy and entropy values of detail coefficients obtained from 13 different types of wavelets that listed in Table 4.2.

Table 4.2. Wavelet Types

<i>Mother Wavelet</i>	<i>Wavelet Number</i>
Daubchies	DB1, DB2, DB4, DB6, DB10
Symlet	SYM4, SYM6
Coiflet	COIF2, COIF5
Bio-orthogonal	BIOR1.5, BIOR2.6, BIOR3.5, BIOR5.5

Figure 4.5 illustrates a Normal beat QRS complex and the signal’s detail coefficients of 4-level DWT decomposition with DB4 wavelet.

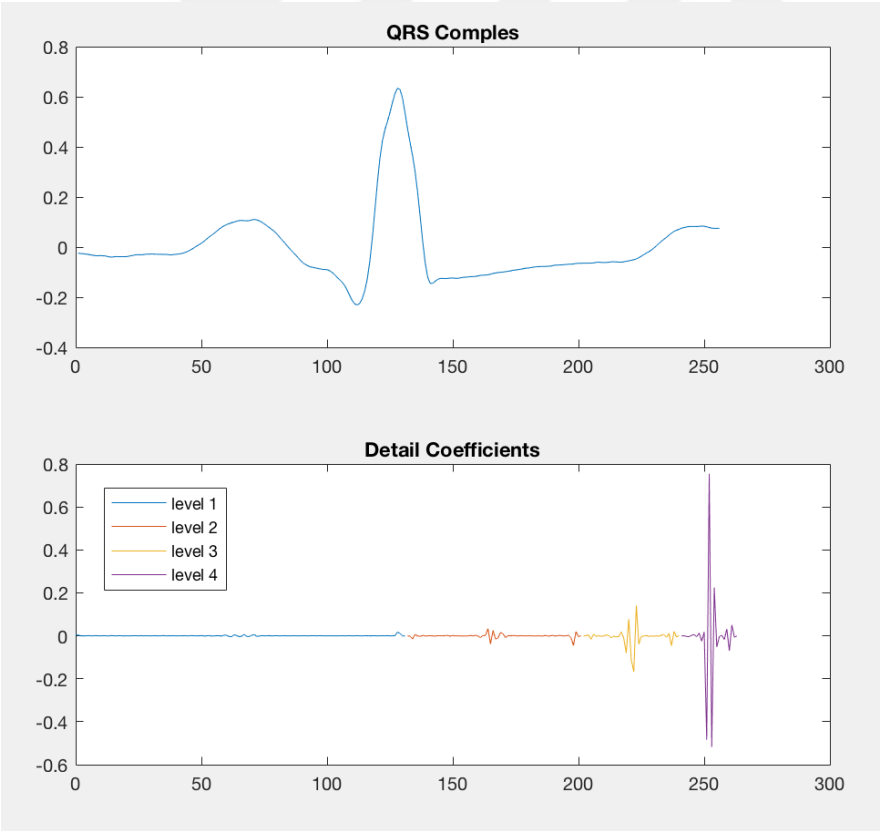


Figure 4.5. Four level DWT Decomposition of A Normal Heart Beat

In MATLAB features extracted from QRS complexes was detected by following steps below and Figure 4.6 shows the block diagram of the algorithm.

1. QRS complexes of each arrhythmia (NORMAL, LBBB, RBBB and PACED) preprocessed peak locations were found for each arrhythmia type with Pan Tompkins Algorithm. Taking 127 sample before and 128 sample after the R samples gave the QRS complex with a length of 256. The number of 7198 ECG beats were obtained from normal, 6902 from LBBB, 6332 from RBBB and 7022 from paced. As a result, combination of each QRS vector creates of a 256x27454 matrix.

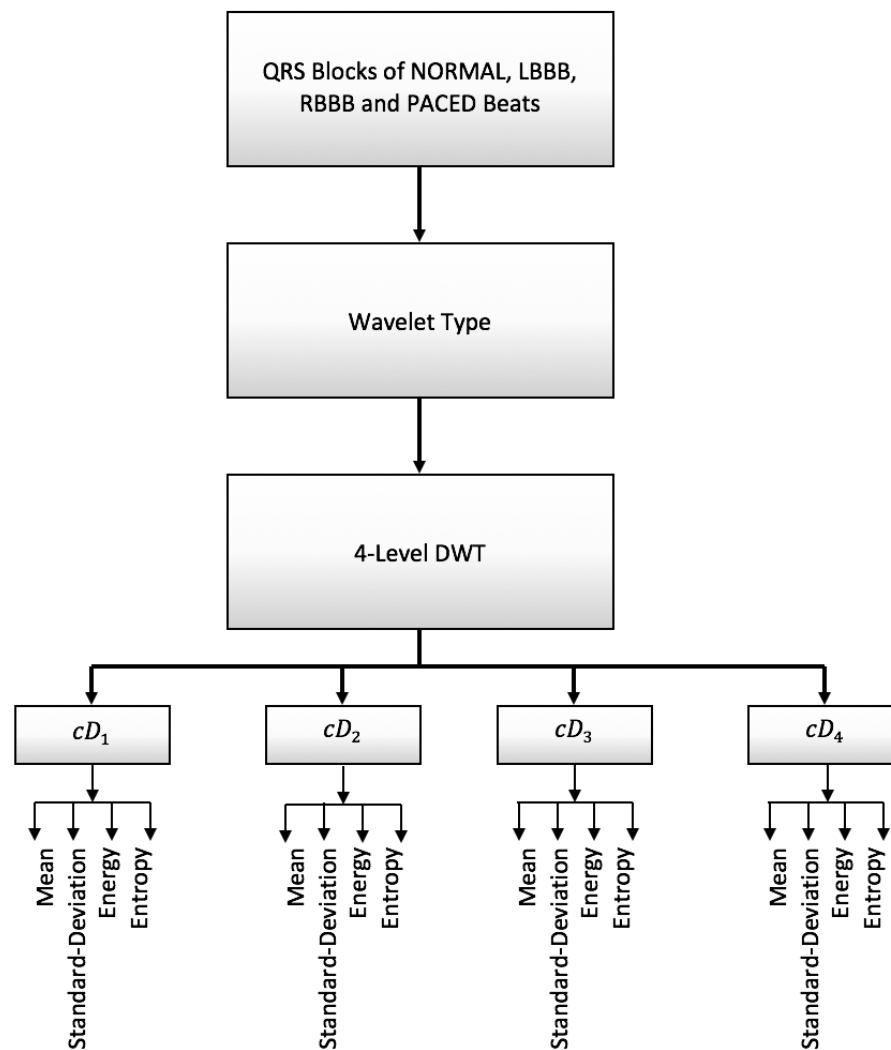


Figure 4.6. Block Diagram of Feature Extraction

2. Each line of the matrix was transformed by DWT and the detail coefficients in 4-levels for each wavelets were generated. Figure 4.6 cD_1 , cD_2 , cD_3 and cD_4 in

orderly represents to detail coefficients of level1, level2, level3 and level4 respectively.

3. Mean, standard-deviation, energy and entropy values of each level detail coefficients were calculated and the result gives 4 features for each level.
4. As a result 16 features extracted from one QRS block for one wavelet type.

In Chapter 3 it was mentioned that “the more the wavelet matches the nature of the signal the more detailed information is obtained from the signal.” For this reason, the detection of best wavelet match for ECG signal among these different wavelets is an indication of the fact that the features obtained from that wavelet type reflect more characteristics of the signal. For this purpose, classification algorithms of NN can be used. Accuracy of the NN illustrates the discriminative behavior of these features. MLP was preferred as NN because of containing the back-propagation technique. In order to obtain a good result in MLP, appropriate training function should be determined. In this thesis, the most successful training function was defined as Levenberg-Marquardt Method in classification of ECG arrhythmias. The comparison bar graph of training functions in MATLAB is shown in Figure 4.7. Features extracted by using DB4 wavelet. Hidden layer number used as default value which is 10. When getting results of the training functions, the only parameter that has changed is the training functions. As clearly shown in Figure 4.7, the function that gives the best accuracy result is Levenberg-Marquardt function.

Equation (26) indicates the accuracy calculation of the classification algorithm. Accuracy rate A is calculated by the ratio of true data in the cluster to all data.

$$A = \frac{TP + TN}{TP + FP + TN + FN} \cdot 100[\%] \quad (26)$$

where TP is the True Positive instances that classified correct, TN is the true negative, FP is the false positive and FN is the false negative instances of the classification .

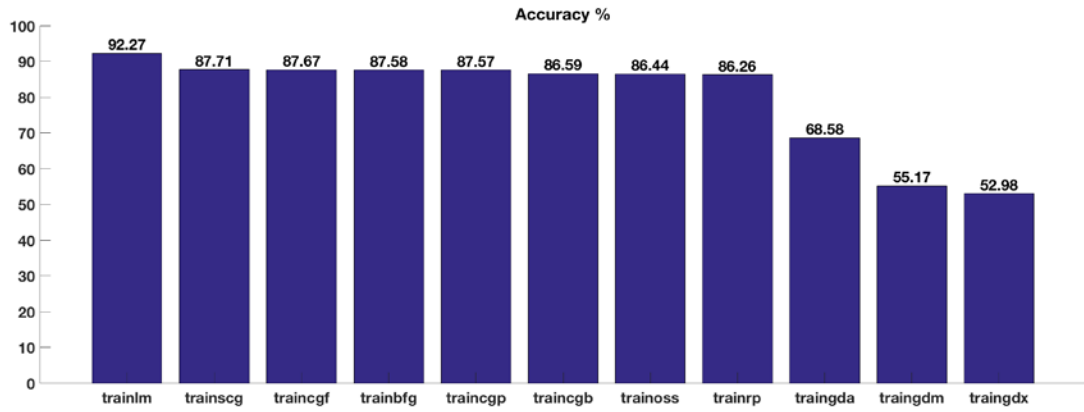


Figure 4.7. Comparison Bar Graph of Training Functions Accuracy Results

Sensitivity (true positive rate) and specificity (true negative rate) of feature extraction and classification are used to calculate the statistical performance of test of the classification. The percentage value of sensitivity is determined by the ratio of the number of true positives to summation of the true positive and false negative. This relationship can be expressed as

$$S_t = \frac{TP}{TP + FN} \cdot 100[\%] \quad (27)$$

The percentage value of specificity is defined by the ratio of the number of true negatives to the true negative and false positive sum. Specificity can be expressed as

$$S_p = \frac{TN}{TN + FP} \cdot 100[\%] \quad (28)$$

After determining training function, the number of hidden neurons, which is suitable for the data set, was tried to be find. Values between 5 and 40 were tested as hidden number by using trainlm function in Matlab. The numbers between 30-32 gave the best result, thus the value 30 is selected.

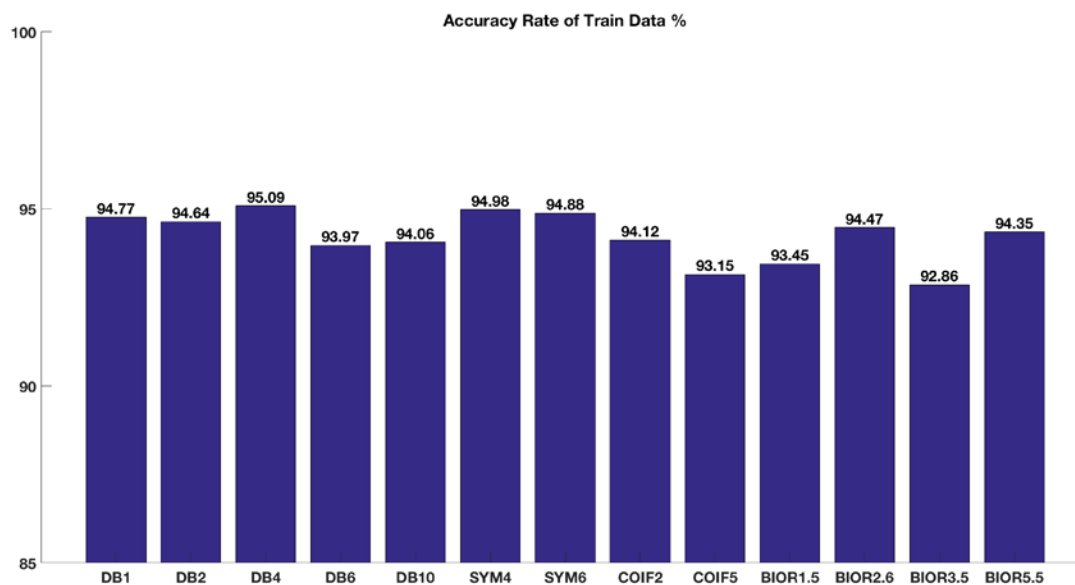


Figure 4.8. Bar Graph of Different Wavelet Type's Accuracy Rates

Table 4.3. Accuracy Rates of Classification of Wavelet Types for Arrhythmias (From Sarvan 2017)

Wavelet Type	Accuracy Rates of Classification Each Arrhythmia			
	<i>Normal %</i>	<i>LBBB %</i>	<i>RBBB %</i>	<i>Paced %</i>
DB1	97,37	95,40	98,08	97,20
DB2	95,68	94,52	97,25	96,65
DB3	95,43	94,27	98,20	95,64
DB4	96,45	95,08	98,14	97,38
DB6	94,71	93,97	96,51	97,00
DB10	94,39	92,96	97,84	96,45
SYM3	94,49	97,83	96,83	96,17
SYM4	95,74	95,21	98,18	96,47
SYM6	96,29	95,80	97,45	96,90
COIF2	95,15	94,03	97,03	96,81
COIF5	95,19	94,70	97,43	96,21

In the Figure 4.8, the accuracy rates of ECG signals classification using 16 features of each 13 different wavelets in 4 levels have been exhibited. As can be seen in that figure, the accuracy comparison of these wavelets are quite high and close. If the results are to be examined in more detail, Table 4.3 gives us some information about why the general accuracy rates are close each other.

The 4 different type ECG signal shapes are being tried to be distinguished. For this reason, a single wavelet may not provide the proper match within these 4 types. Table 4.3 specifies the classification accuracy ratios for these 4 different ECG signal. The wavelet DB1 gives higher accuracy for Normal rhythm and also has highest average accuracy value for all classes. DB3 and SYM4 wavelets for RBBB rhythm, SYM3 wavelet for LBBB rhythm and DB4 for paced arrhythmia has higher accuracy when the Table 4.3 is considered.

Considering the obtained results in Table 4.3; DB3, DB4, SYM3 and SYM4 wavelets have been combined with DB1 in order to increase the accuracy. 16 features are extracted from each wavelet and combination of two wavelets provide 32 features. 27454 instances of 32 features were trained and tested using 30 hidden neurons in the Neural Network with Levenberg-Marquardt function. It has been observed that in training result of Neural Network, the features from multi-wavelets provide higher degree of accuracy than the individual use of wavelets. Table 4.4 is a comparison table between single wavelets and the combination with DB1. It illustrates results of statistical calculation percentages, which are sensitivity, specificity, positive prediction, negative prediction and accuracy of neural network (Sarvan and Özkurt, 2017).

The results brings to mind the following question: “Can the combination of the features obtained from different wavelets different levels give better result?”. Next section examines the answer of this question. Because since human health is the topic, the best must be find.

Table 4.4. Neural Network Classification Results of Feature Sets (From Sarvan 2017)

Features from Wavelets	Sensitivity %	Specificity %	Positive Predictive %	Negative Predictive %	Accuracy %
16 Features with DB1	98,40	95,62	95,45	98,46	96,95
16 Features with DB4	98,18	95,07	94,86	98,25	96,55
16 Features with SYM4	97,97	94,54	94,30	98,06	96,18
16 Features with SYM3	97,81	94,14	93,84	97,90	95,87
32 Features with DB1 & DB4	99,45	98,41	98,39	99,45	98,92
32 Features with DB1 & SYM3	99,38	98,20	98,18	99,39	98,78
32 Features with DB1 & SYM4	99,30	97,99	97,96	99,31	98,63

4.4. Feature Selection

Features that have high discrimination ability is an important issue for classification algorithms. In this thesis features were selected by using genetic algorithms. Feature set have been prepared by considering 13 wavelets (Table 4.2). Each wavelet gives 16 features and as a result 208 features were included in the selection set. Main purpose is to find best features for each ECG arrhythmia class. In this section two methods will be described as feature selection. One of is selecting features with GA that written as a Matlab algorithm and the other one is selecting features with multi-objective genetic algorithm (MOGA) that is developed in Java.

4.4.1. GA Implementation with Matlab

GA is designed in Matlab for feature selection. In this section, GA results depend on the following parameter settings:

1. *Representation:* Binary-coded representation
2. *Crossover Probability:* 0.7
3. *Mutation Probability:* 0.05

4. *Mutation Method:* Bit-swap
5. *Parent Selection:* Tournament Selection
6. *Elitism:* Yes

Proposed GA based feature selection was used binary system representation for individuals. Binary (the combination of 1 and 0s) strings were represented both genotype and phenotype. Selected level wavelet is denoted as “1” and other levels are set to “0” in genotype structure which is shown in Figure 4.9. Phenotype shows which features are selected for each level. A phenotype that has length of 208 was obtained from a genotype which has the size of 52. In fitness function part, phenotype has been operated. Genotype includes 4 wavelet levels fixedly. Thus, the size of phenotype was 16 since there are 4 features.

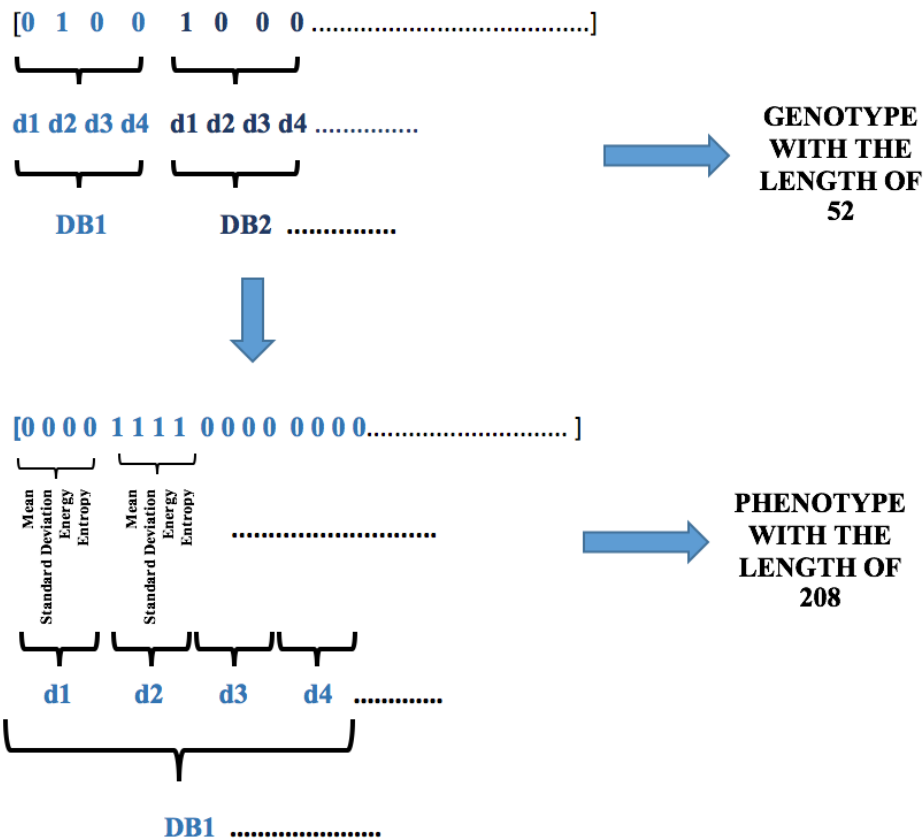


Figure 4.9. Genotype and Phenotype Representation of GA (From Sarvan 2017)

Two-level tournament selection was realized in randomly formed initial population. Number of the randomly chosen individuals has been divided into two subgroups in its own. These subgroups were competed with respect to outputs of fitness function and the winners were fed into final pool. In order to form a new individual from the

parents chosen from the final pool, crossover has been applied. Crossover method rates were specified as; 0.7 uniform crossover, 0.2 two-point and 0.1 single-point and roulette wheel selection has been used. Two individuals, each produced from crossovers, were thrown into the pool of new population. Elitism has been carried out between the new and the old populations. The fitness value of the best individual of the old population was compared with the fitness value of the best individual in the new population and was retained in the new population. Thus, the best individual found was kept in the population during all generations. After the change of elitism, dice was rolled subject to mutation rate while keeping the best individual inside. If the dice is less than the mutation rate, mutation is performed by applying bit-swap mutation to the individual, else mutation is not performed. In this thesis, individuals in the population can mutate with a probability of 5% (Sarvan et al., 2017).

High performance rate is expected from NN test results for classification arrhythmia types of an ECG signal. Therefore, fitness function designed associatively with accuracy rate of the classifier. 16 features was selected from features pool according to phenotype of the individual. Instances of these features were given as input to the NN and trained with Levenberg-Marquardt (trainlm) function. Input data was randomly divided as 70% train data, 15% test data and 15% validation data. The network was trained and tested with default value of 10 hidden neurons. Accuracy of the test and validation data have been obtained from NN by using equation (26).

The fitness value of the GA was related to accuracy results of Normal (A_{Normal}), LBBB (A_{LBBB}), RBBB (A_{RBBB}) and Paced (A_{Paced}) signals. Taking the average value of test and validation accuracies of these four class gives the fitness function in equation (29).

$$f_i = \frac{A_{Normal} + A_{LBBB} + A_{RBBB} + A_{Paced}}{4} \quad (29)$$

where f_i denotes the fitness function value of the i th individual of current population. Result of the fitness function was taken as fitness value of each individual in population and tried to maximize in GA.

4.4.1.1. System Architecture of GA

Architecture of the GA is indicated in Figure 4.10 and according to system architecture, the algorithm is described in the following steps.

- 1) *Initial Main Data Set of Features.* At this step, a 208x27454 main feature data set was obtained using DWT from 4 different types of arrhythmia samples taken from the MIT-BIH database.
- 2) *Population.* Population initialized as random.
- 3) *Parent Selection.* The tournament selection method was used to select the parents which were used in crossover.
- 4) *New Individuals.* As a result of crossover, new individuals were created. Applied the mutation to these individuals according to mutation probability ratio.
- 5) *New Population.* The individuals that were created in step 4 were formed a new population. The old population was replaced by the new population.
- 6) *Conversion process of genotype to phenotype.* This step converts each parameter and feature chromosome into a phenotype from the genotype.
- 7) *Feature Selection from Main Data Set.* After conversion of genotype to phenotype, 16 features selected from the main feature data set were parsed.
- 8) *Division of Selected Features.* The sub-feature data set was divided for use in the NN as 70% training, 15% testing and 15% verification data.
- 9) *Neural Network.* The test and verification data set is used to calculate the NN classification accuracy by using Levenberg-Marquardt training function.
- 10) *Fitness Function Evaluation.* When the classification accuracy is obtained, each individual is evaluated by the fitness function result.
- 11) *Is Termination criteria reached?* If the termination criteria is verified, it is returned to the best set of features that found; otherwise the genetic operation will continue to operate from the next generation. The termination criteria in this study was set as the number of generations assigned. When the given number of generations is completed, the termination criteria is verified.
- 12) *Return Best Feature Subset.* The best set of features found throughout the generations was returned.

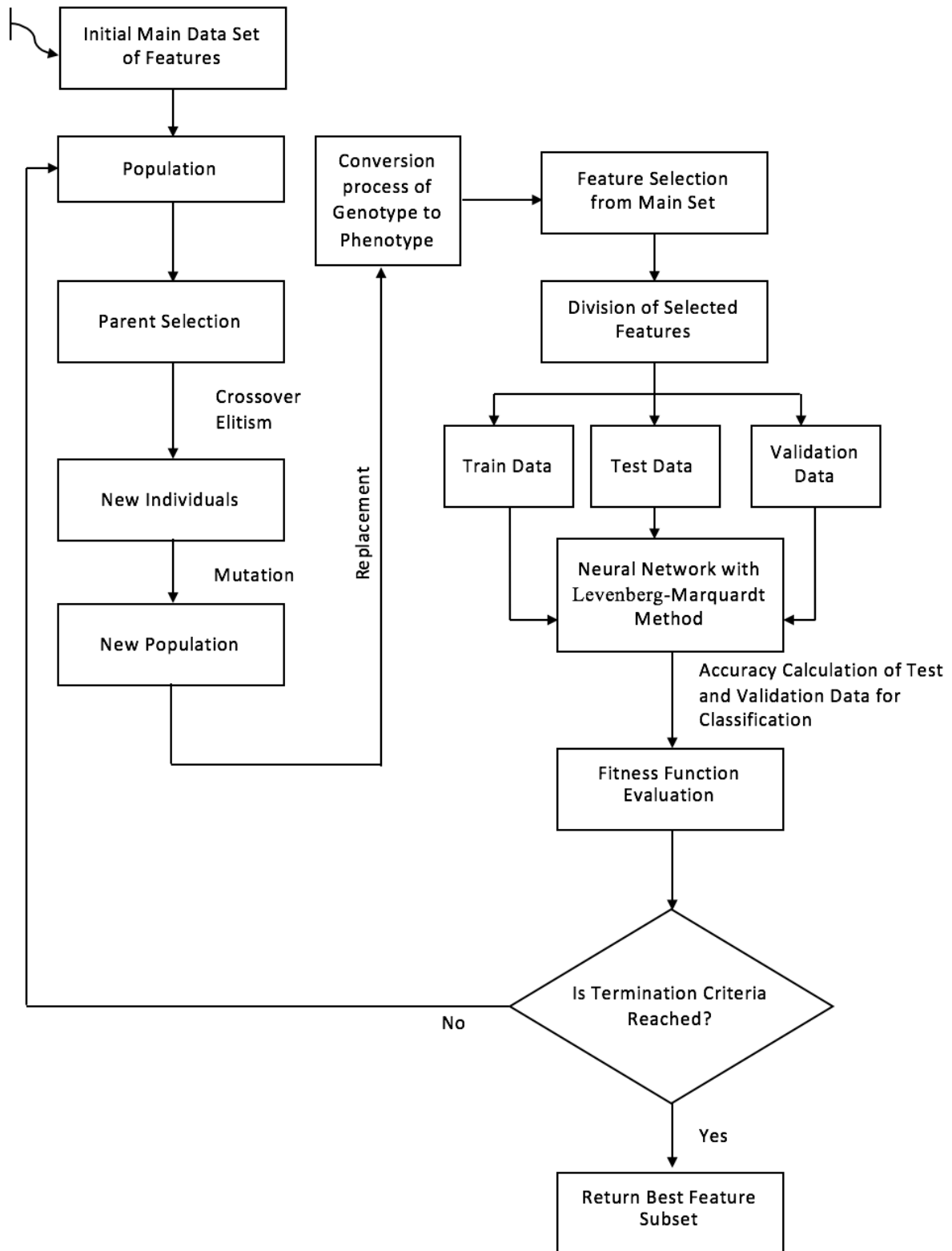
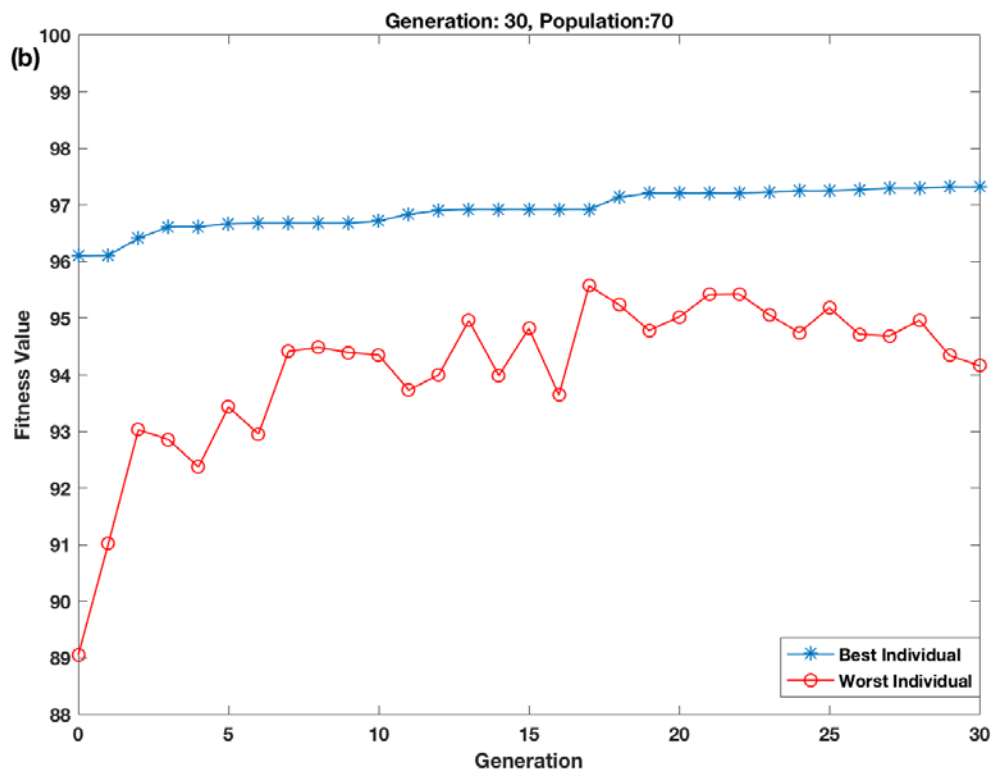
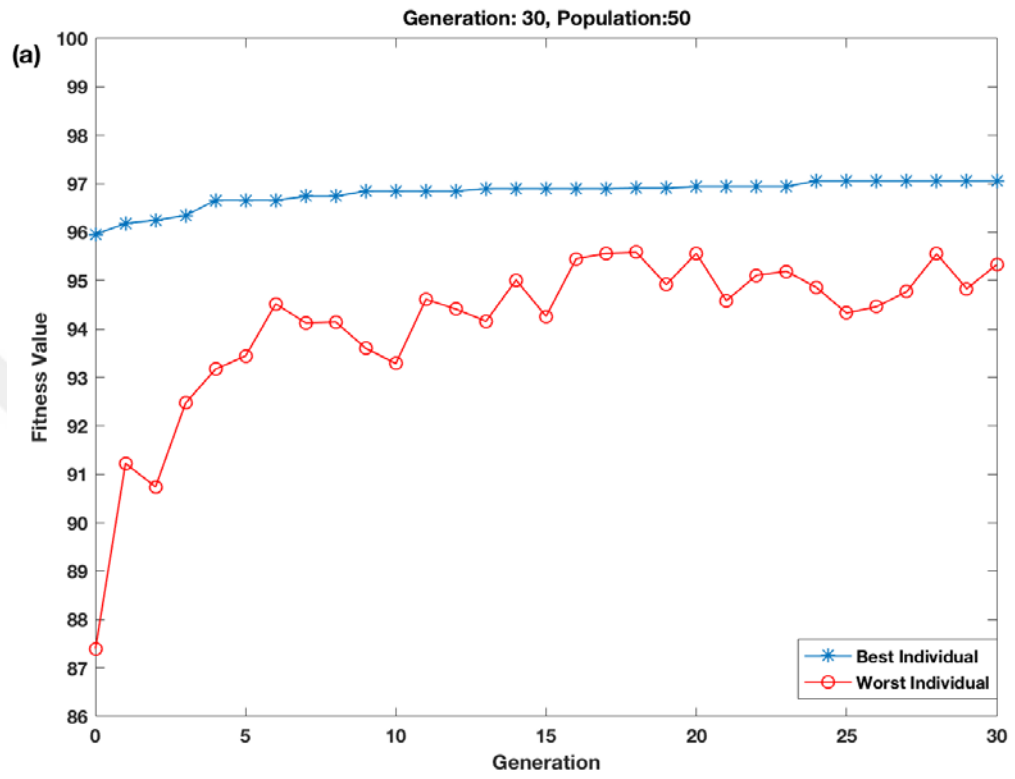


Figure 4.10. Architecture of GA

4.4.1.1.Results

The Matlab-based GA obtained a feature subset, that contains 16 features, from the pool of 208 feature set by trying to maximize the fitness function value.



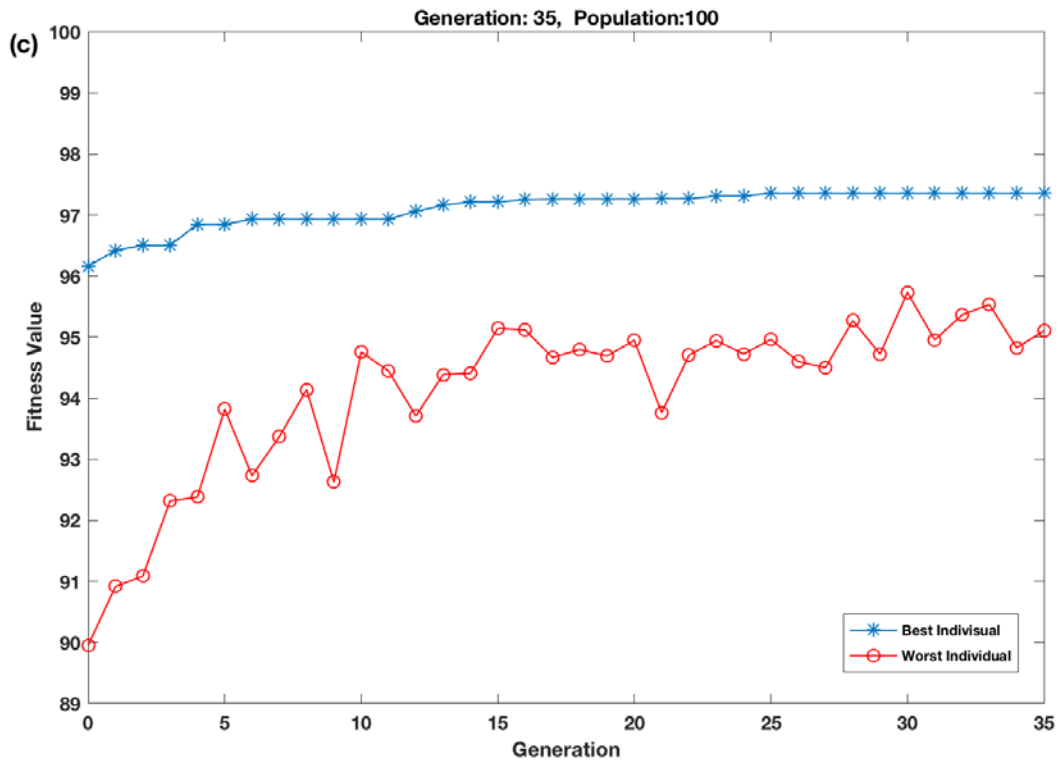


Figure 4.11. Graph of Fitness Function Values of Generations; (a) 30 generations with 50 population (b) 30 generations with 70 population (c) 35 generations with 100 population

Figure 4.11 shows the comparison of fitness value of the best and worst individuals in each generation by changing number of population and generation. When Figure 4.11.(a)-(c) are examined, the fitness of best individuals produced through out the generations has increased.

As a result of three different generations and population experiment, the best and worst subset of the features are illustrated in Table 4.5. When the Table 4.5 is examined, sub feature set that consisting of coefficients of DB2, BIOR1.5, BIOR 2.6 and BIOR3.5 has been identified as the most successful feature set for both 30 generations with 70 individuals in population and 35 generations with 100 individuals in population. Seed values of each run is also given in the Table 4.5.

Table 4.5. GA Results of Three Different Run (From Sarvan 2017)

GA Results	Generation: 30 Population: 50	Generation: 30 Population: 70	Generation: 35 Population: 100
Best Individual Fitness Function %	97,06	97,32	97,36
Best Feature Set	DB2 (cD_1) + COIF2(cD_4) + BIOR1.5 (cD_2) + BIOR2.6 (cD_4)	DB2 (cD_1) + BIOR1.5 (cD_1) + BIOR2.6 (cD_4) + BIOR3.5 (cD_4)	DB2 (cD_1) + BIOR1.5 (cD_1) + BIOR2.6 (cD_4) + BIOR3.5 (cD_4)
Worst Individual Fitness Function %	87,39	89,05	89,95
Worst Feature Set	DB4 (cD_2) + SYM4(cD_1) + COIF5(cD_1+cD_2)	DB2 (cD_2) + COIF2 (cD_1) + COIF5(cD_2) + BIOR5.5 (cD_2)	SYM4 (cD_2) + COIF2 (cD_1) + BIOR3.5 (cD_3) + BIOR5.5 (cD_2)
Seed Value	1408513609	1406764815	1411520240

4.4.2. Multi-Objective GA Implementation with JAVA

The selection of the most suitable feature subset among the obtained features from 13 wavelets was determined by using the multi-objective approach with the GA structure. The GA structure was constructed in the Java programming language using the "Watchmaker Framework". The Watchmaker Framework is an object-oriented framework developed in the Java programming language that is platform independent, with the goal of implementing evolutionary and genetic algorithms. It is a freely available open source software with Apache Software License (Watchmaker Framework for Evolutionary Computation, 2017). And the other software that used in this section is Weka application program interface. Weka was developed by the University of Waikato for data mining tasks. It includes machine learning algorithms. Algorithms in the Weka library can be called and implemented in java code. For this reason, Weka's MLP algorithm was used for the neural network structure in the fitness function part (Machine Learning Group at the University of Waikato, 2018).

GA is designed in Java by using Watchmaker framework with the following parameter settings:

1. *Representation*: Bit-string
2. *Crossover Probability*: 0.85
3. *Mutation Probability*: 0.05
4. *Mutation Method*: Bit-swap
5. *Parent Selection*: Tournament Selection
6. *Elitism*: No

In this section, it is aimed to determine the minimum number of features with low error rate and high accuracy ratio. These 3 different values were reduced to a single scale and a multi-objective scalar approach was applied (Sarvan and Özkurt, 2018).

In the fitness function, the main data set consisting of 208 features was given to the NN as 7198 NORMAL, 6902 LBBB, 6332 RBBB and 7022 PACED rhythms. The input data was divided into 70% training and 30% test data. In MLP algorithm parameters were set as in Table 4.6. In Table 4.6 learning rate is a parameter of back-propagation algorithm and small learning rate provide stabilized weight approaches by obtaining averages of past inputs. Higher learning rates are preferred for fast adaptation according to the actual changes in distribution of the process. Learning rate can be greater than 0 and equal or lower than 1. The small value of 0.1 was set as learning rate according to this application requirement. Momentum is the balancing parameter used for not to slow down the learning rate between high and low eigenvalues for the back-propagation algorithm. It takes a decimal value between 0 and 1. ‘0.2’ ratio was set because it corresponds to the general value 1/5 rule according to parameter tuning. Hidden layers set as the default value 10.

Table 4.6. Parameters of MLP for MOGA

<i>Parameter</i>	<i>Value</i>
<i>Learning Rate :</i>	0.1
<i>Epochs :</i>	1000
<i>Hidden Layers :</i>	10
<i>Momentum :</i>	0.2

The fitness function was designed to obtain a single value by looking at three different

criteria: average accuracy rate of classes, the root mean square error (RMSE) of the classification, and the number of features selected by the GA. The accuracy rate of the test data obtained from the MLP algorithm was calculated according to the equation (26) and Average accuracy rate A_{avg} can be calculated as

$$A_{avg} = \frac{A_{Normal} + A_{LBBB} + A_{RBBB} + A_{Paced}}{4} \quad (30)$$

where A_{Normal} , A_{LBBB} , A_{RBBB} and A_{Paced} are the accuracy results of the Normal, LBBB, RBBB and Paced signal classes. The other criteria of RMSE value was calculated as equation (31) by depending on MLP results.

$$E_{RMSE} = \sqrt{\frac{1}{N} \sum_{i=1}^N (d_i - \mathbf{w}x_i)^2} \quad (31)$$

Where d_i denotes the target of i th instance, \mathbf{w} is the weight vector and x_i is the i th input value. Multiplication of weights and input value gives the actual output of the classification algorithm. Subtracting this value from target provide the error.

The fitness function obtained by using the criterion results, which obtained from equation (30) and (31), can be denoted as

$$f_i = \sqrt{(100 - A_{avg})^2 + N^2 + E_{RMSE}} \quad (32)$$

where f_i represents the fitness value of i th individual of current population. The average accuracy rate obtained from the classification (A_{avg}) was subtracted from 100 and squared. Squared value of number of selected features (N) added to this value and normalized by taking the square root. As a final term, the RMSE (E_{RMSE}) of the classification was added. Thus, minimizing the number of errors and the number of features was targeted.

4.4.2.1. System Architecture of Multi-Objective GA

Architecture of the MOGA is shown in Figure 4.12 and system architecture of the java is used the following steps.

- 1) *Initial Main Data Set of Features*. At this step, a 208 features with 27454 instances was extracted from the signals Normal, LBBB, RBBB and Paced by using DWT. The instances separated with randomized partition as 70%

train and 30% test data. The number of 19218 instances used to train the MLP and remaining 8236 instances used for test the trained MLP.

- 2) *Conversion process of genotype to phenotype.* Genotype contains 52 binary data. 1 refers the selected level of detail coefficients and 0 refers non-selection. 4 feature calculated by using each level of detail coefficients. This means binary values folds to 4 by depending on corresponding detail coefficients. Phenotype reflects genotype influence. As a result, it includes 208 binary data. This step converts the 52-length genotype into a 208-length phenotype,
- 3) *Feature Selection from Train Data Set.* After conversion of genotype to phenotype, N features selected from the train data set.
- 4) *Feature Selection from Test Data Set.* Same as step 3. Same N features selected from the test data set.
- 5) *Neural Network.* MLP was trained with the selected features and tested. Accuracy and RMSE value was calculated from the MLP to use an object of fitness function.
- 6) *Fitness Function Evaluation.* Fitness value of the individual was calculated by using three parameters; Selected feature number N , accuracy rate and RMSE value of MLP testing results. Each individual was evaluated by the fitness value.

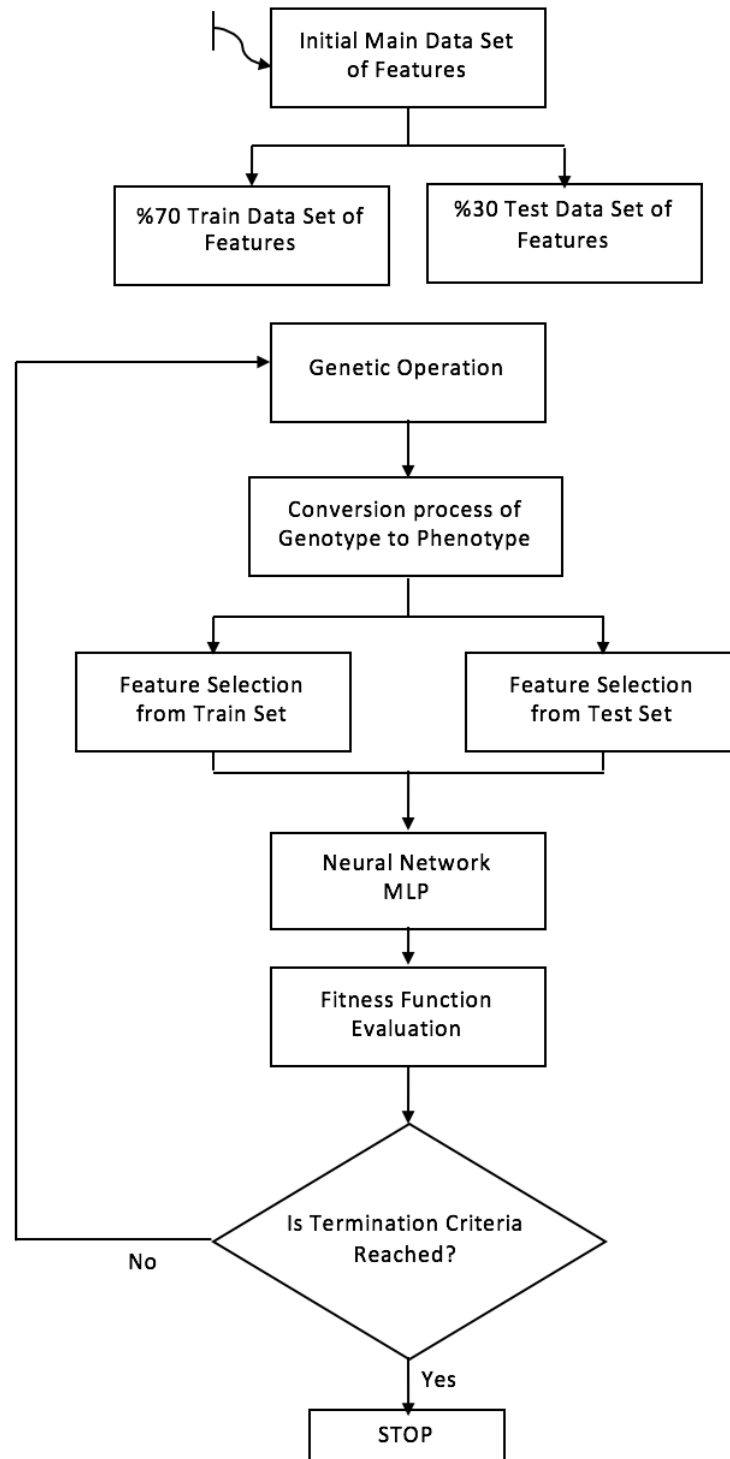


Figure 4.12. Architecture of GA with Multi-Objective Approach

- 7) *Is Termination criteria reached?* Checks the termination criteria, which was the limited number of generations. If the program reaches to defined generation number, then the program stops otherwise the genetic operation will continue to operate from the next generation.

- 8) *Genetic Operation*. This contains same algorithm structure of standard GA. Respectively follows population, parent selection, new individuals and new population steps with defined parameters such as crossover, mutation in this section.

4.4.2.2. Results

The Java-based GA with multi-objective approach was used 27454 instances that is obtained from MIT-BIH database. These instances were separated 19218 instances as train and 8236 instances as test before the starting algorithm.

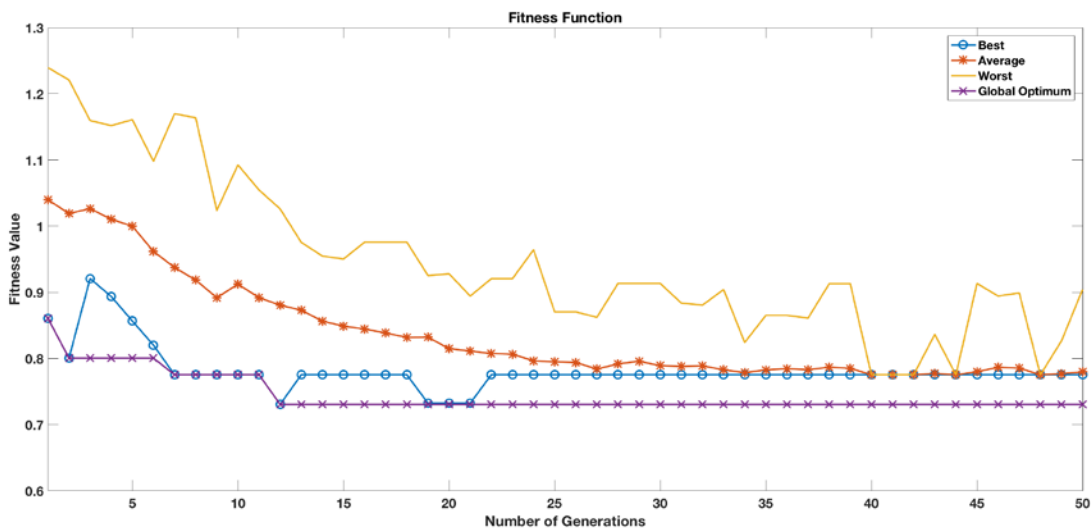


Figure 4.13. Fitness Function with Respect to Generations

The Figure 4.13 shows the fitness function graph of best, average, worst and “global best” individuals of each generation. The value of fitness function has been tried to be minimized along the generations. In this experiment, the number of generation was selected 50 and the number of individuals in the population was chosen 30. The reason of these population and generation parameters are reduction of diversity. If the graph in Figure 4.13 has been checked, the diversity of generations has been reduced after the 22nd generation and the structure remains in the local optimum and has not been able to catch the global optimum again. The best, the worst, and the average values of the fitness function of individuals were superimposed on the graph. This shows that the individuals were alike. For this reason, the increase or decrease in the number of generations would not be effective on the performance. Different methods such as implementing distributed island model in GA, have to be tested to increase diversity to probability of finding better solutions.

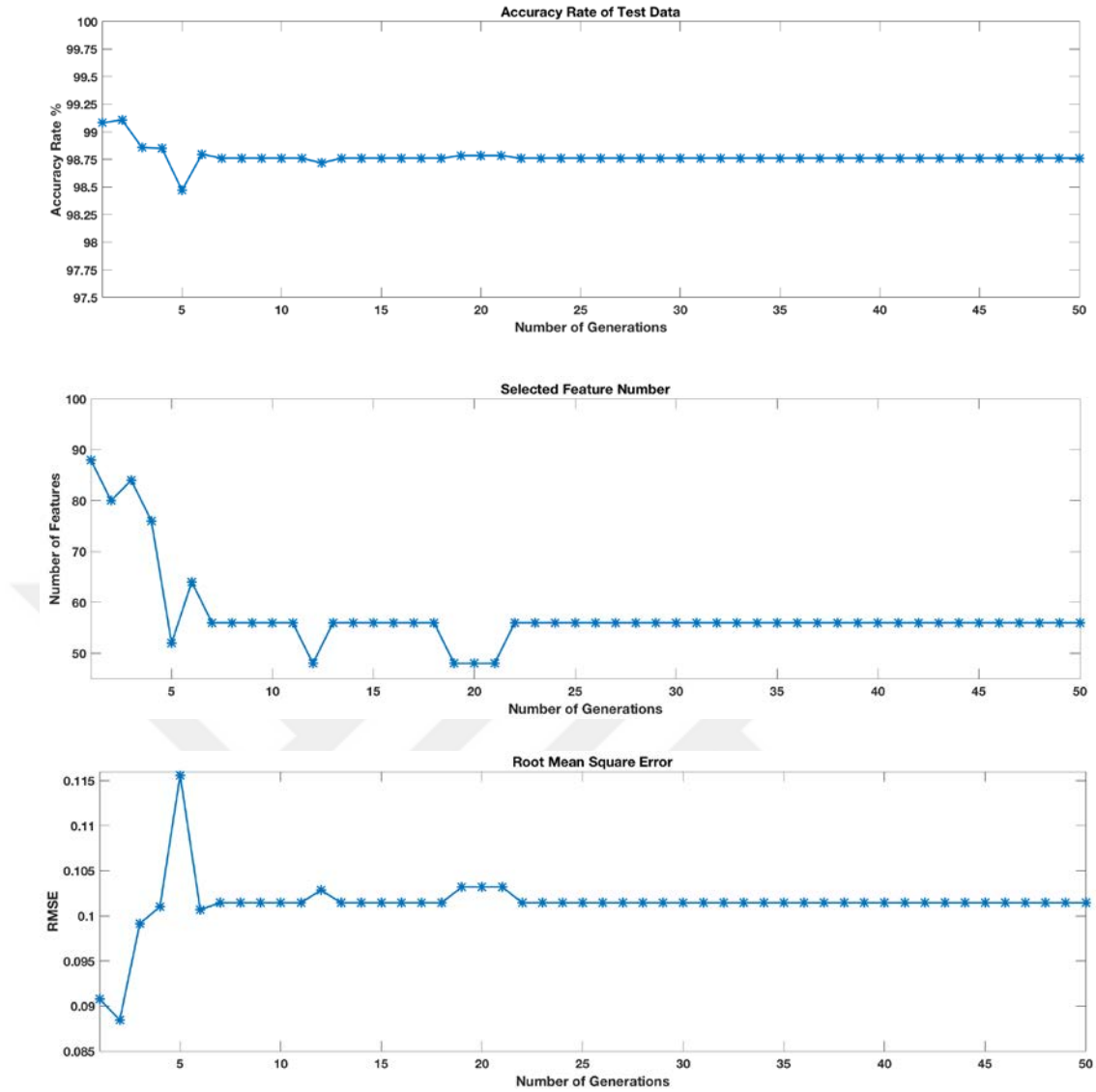


Figure 4.14. Multi-Objective Approach Criteria with Respect to Generations

The best global individual that found along the generations includes 48 features. Figure 4.14 indicates the fitness function objects separately for the best individual of each generation. When the number of features were reduced, the rates of accuracy and RMSE were balanced at a certain value according to graphs in Figure 4.14.

As a result, the Java-based GA with multi-objective approach found 48 features with 98,79% accuracy rate by trying to minimize the fitness function (Sarvan and Özkurt, 2018). Below list contains these 48 features;

- $DB1(cD_3)$
- $DB6(cD_2+cD_4)$
- $DB10(cD_1)$
- $SYM6(cD_1)$
- $COIF5(cD_2+cD_3+cD_4)$
- $BIOR2.6(cD_3+cD_4)$
- $BIOR3.5(cD_1)$
- $BIOR5.5(cD_4)$

4.5. Summary of Arrhythmia Detection

Highly discriminative features that provide high accuracy ratio in classification algorithms have been tried to be determined by using different methods. The review is chronologically listed in below.

1. The accuracy ratios of the 16 features obtained from a single wavelet were examined.
2. Four wavelets with highest accuracy ratio was taken and 32 features extracted from combination in pairs of these wavelets.
3. Compared to use of single wavelet, the combined wavelet use of two wavelets achieved a higher accuracy.
4. The interactions of features from different wavelet's different levels were examined. For instance, by using the combination of features that obtained from the fourth-level detail coefficients of DB4 and second-level detail coefficients of SYM4.
5. GA was designed to find 16 features from 208 feature pool by combining 4 features that acquired from single-level detail coefficients of a wavelet. The important point was to find high discriminative features so the fitness function was determined as the accuracy rate of the neural network.
6. Besides the accuracy rate, the error of neural network was counted to the application by using multi-objective approach. Main idea was to find the minimum number of features with the lowest classification error and highest accuracy rate. These three objects were added to fitness function. The GA with multi-objective approach found 48 features with 98,79% accuracy rate.
7. In step 5 and 6 describe EA based methods that randomly generate combination of features by selecting from feature pool. In both methods, the number of generations was assigned as termination criteria and GA returns the best solution it has found.

4.6. JAVA Implementation of Arrhythmia Detection

As a result, of these steps, a simulation program designed in JAVA. In Table 4.7 shows the method name and best features that obtained from the method. These features were used in simulation program to classify ECG heart beats. In Table 4.7 cD_1 to cD_4 refers

to wavelet detail coefficients of n th level. Mean, standard-deviation, energy and entropy values were calculated from detail coefficients and these values were created 4 features for each level. The simulation program contains a selection combo-box on master window. The user can select the method and the features which were obtained by selected method, are extracted from the simulated signal. A division was made between the records that approximately 80% of the records were used for train to MLP and 20% of them were used in simulation program. Therefore, Table 4.7 contains MLP training accuracy results. Hidden layer number was selected as default value of 10, learning rate was set 0.1 and momentum of MLP was set as 0.2. Each run of the table used same MLP parameters. Train results saved in order to be included in the simulation.

Table 4.7. Feature Set of Three Methods

Method	Features	Number of Features	MLP Training Accuracy Rate %
Combination of two wavelets	<ul style="list-style-type: none"> ➤ DB1 ($cD_1+cD_2+cD_3+cD_4$) ➤ DB4 ($cD_1+cD_2+cD_3+cD_4$) 	32	95.89
Selected features from GA	<ul style="list-style-type: none"> ➤ DB2 (cD_1) ➤ BIOR1.5 (cD_1) ➤ BIOR2.6 (cD_4) ➤ BIOR3.5 (cD_4) 	16	92.77
Selected features from GA with multi-objective approach	<ul style="list-style-type: none"> ➤ DB1(cD_3) ➤ DB6(cD_2+cD_4) ➤ DB10(cD_1) ➤ SYM6(cD_1) ➤ COIF5($cD_2+cD_3+cD_4$) ➤ BIOR2.6(cD_3+cD_4) ➤ BIOR3.5(cD_1) ➤ BIOR5.5(cD_4) 	48	96.69

Table 4.8. Simulation Data Detail of the Software

NORMAL		
Total: 40 files and 1200 seconds		
MIT-BIH Record No	File Count	Time Duration of one File
100	10	30 Seconds
101	10	30 Seconds
105	10	30 Seconds
112	10	30 Seconds
LBBB		
Total: 38 files and 1140 seconds		
MIT-BIH Record No	File Count	Time Duration of one File
109	10	30 Seconds
111	10	30 Seconds
207	8	30 Seconds
214	10	30 Seconds
RBBB		
Total: 39 files and 1170 seconds		
MIT-BIH Record No	File Count	Time Duration of one File
118	9	30 Seconds
124	10	30 Seconds
212	10	30 Seconds
231	10	30 Seconds
PACED		
Total: 40 files and 1200 seconds		
MIT-BIH Record No	File Count	Time Duration of one File
102	10	30 Seconds
104	10	30 Seconds
107	10	30 Seconds
217	10	30 Seconds

In Table 4.8 shows the list of simulation signal files. Each record has 30 seconds time interval.

The multi-thread structure was applied when the program was created. While drawing the signal on the screen, simultaneously extracting the features from the signal and classifying unlabeled signal using the trained MLP according to the extracted features. Algorithm of simulation contains following methods; “Get Data” that reads txt files from simulation folder, “Convolution”, “Pan Tompkins”, “DWT Decomposition”, “Features” that contain mean, standard-deviation, energy and entropy calculations. High and low pass filter coefficients of DWT was obtained from Matlab (was given in Appendix 1) and saved to a text file.

Figure 4.15 shows selection of the technique that features obtained.



Figure 4.15. Selection of Feature Set from Combobox

After selecting technique (Figure 4.15), choosing a signal that belongs to an arrhythmia class (Figure 4.16).

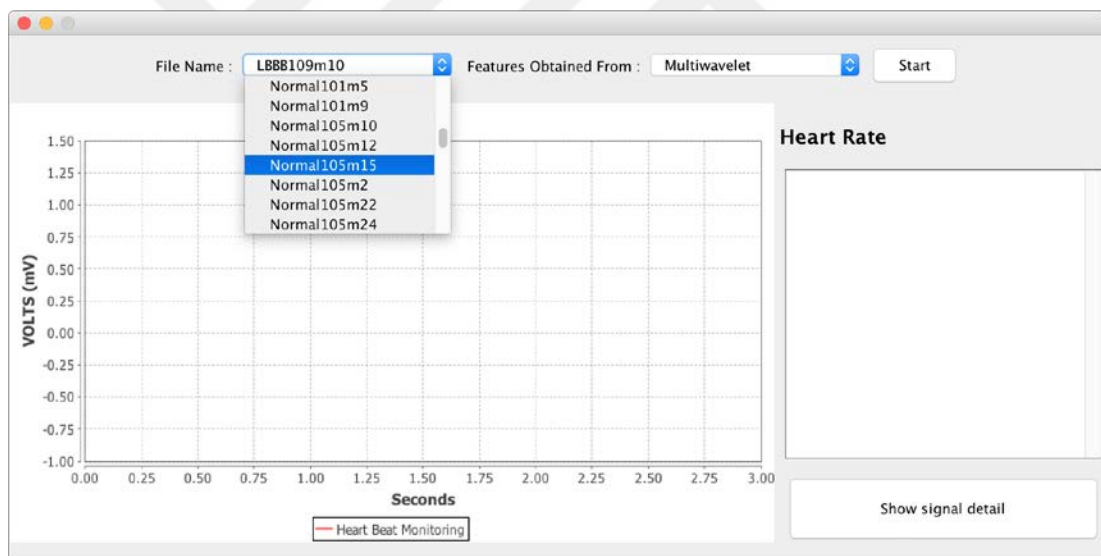


Figure 4.16. Selection of ECG Signal Type

Click the start button, then program starts to draw the signal by reading corresponding text file. While the signal is drawing on the screen, classification decision of the trained MLP is listed for each beat of the signal on listbox of the program.



Figure 4.17. Simulating Signal and Labelling the Beats

Each beat can be selected from the listbox and displayed with the help of a separate window to examine more detail information. Figure 4.18 shows 28th beat of a Normal signal that classified by using multiwavelet features.

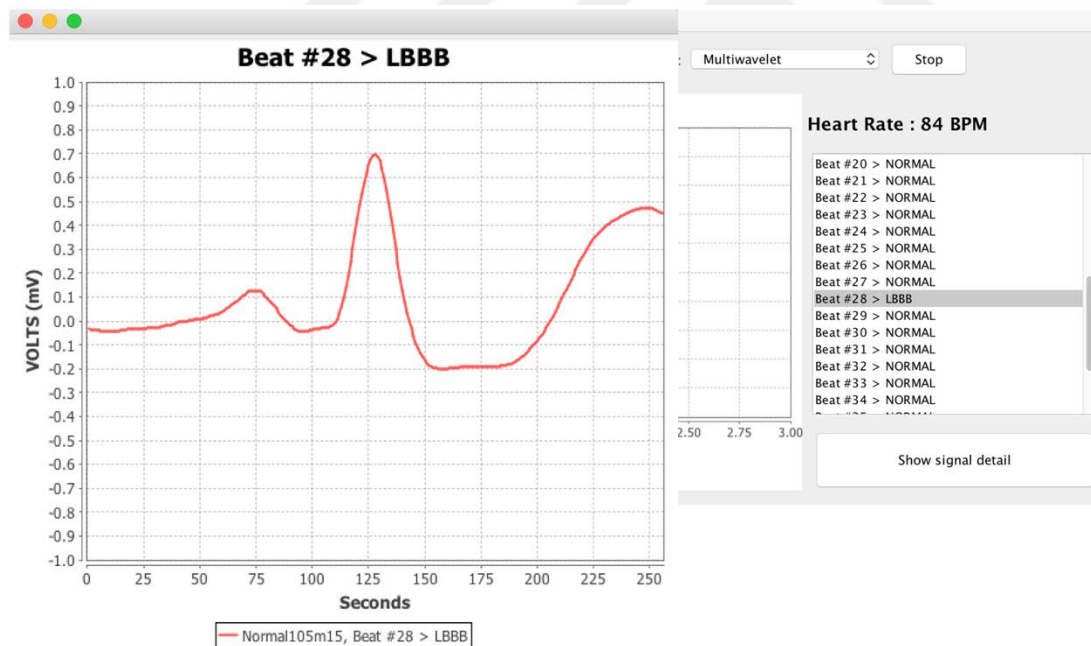


Figure 4.18. Pop-up Window for Beat Analysis in Multiwavelet

Figure 4.19 shows 28th beat of a Normal signal that classified by using selected features by GA with multi-objective approach.

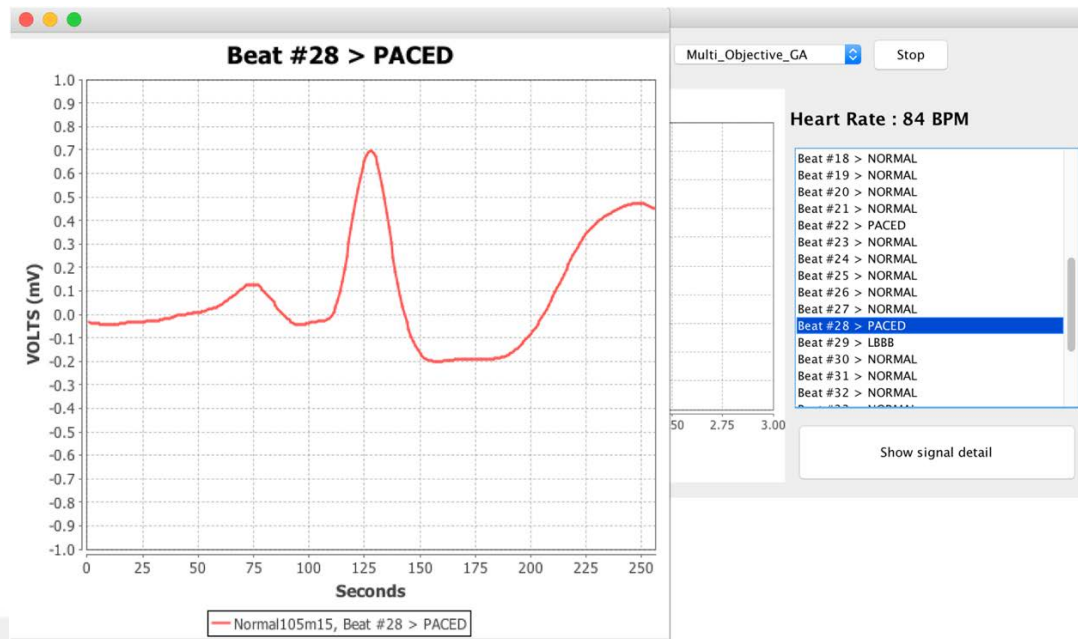


Figure 4.19. Pop-up Window for Beat Analysis in MOGA

4.7. Discussion

Best features for ECG signal types of Normal, LBBB, RBBB and paced tried to be determined. Three different techniques were used; Combination of two wavelets, selecting features from a feature set by using GA and selecting features from a feature set by using GA with multi-objective approach. Each method results trained with MLP according to same parameters. A simulation program designed to compare and observe the techniques on a signal. It was attempted to observe how a trained MLP with found features would result in real-time signal use.

As a result of each feature set that was taken from Normal rhythm records, provides high accuracies on simulation part which is shown in Table 4.9. Selected features with GA and Multiwavelet features give the best classification result for Normal signals. However, the record 105 has worst accuracy result for each method. According to MIT-BIH database record no 105 contains a property that the record has noises and artifacts.

Table 4.9. Simulation Results of Normal Rhythm Records Classification

Method	Record No	Normal Beats	LBBB Beats	RBBB Beats	PACED Beats	Correctly Labelled Instances %
MultiWavelet	100	368	0	3	0	99,19
GA		369	0	2	0	99,46
MOGA		366	0	5	0	98,65
MultiWavelet	101	304	1	1	4	98,07
GA		300	1	6	3	96,77
MOGA		294	0	12	4	94,84
MultiWavelet	105	342	51	7	39	77,9
GA		329	22	3	85	74,94
MOGA		339	58	4	38	77,22
MultiWavelet	112	371	1	5	28	91,61
GA		384	5	7	9	94,81
MOGA		375	1	0	29	92,59

Table 4.10 shows the simulation results of LBBB rhythms. Generally Multiwavelet and selected features from GA with multi-objective approach provide the high classification results. Record 207 has the lowest accuracy rates. When MIT-BIH arrhythmia database is controlled, record 207 has a special note. It was written that the record is hard to labelled in a class. Most of these record beats are AV block and LBBB. But the conduction block occasionally changed to a right branch block pattern. Unidentified beats may have been placed under the paced beat in the classification for this reason. This may have resulted in a lower accuracy rate for the record 207 than other records.

RBBB classification results of the simulation is illustrated in Table 4.11. Almost each three method' s features give the good results. The beat numbers of each RBBB record has been parsed according to the results of the MLP labelling that is shown in Table 4.11.

Table 4.10. Simulation Results of LBBB Records Classification

Method	Record No	Normal Beats	LBBB Beats	RBBB Beats	PACED Beats	Correctly Labelled Instances %
MultiWavelet	109	7	407	1	0	98,07
GA		6	399	5	5	96,15
MOGA		6	406	1	2	97,83
MultiWavelet	111	5	335	1	6	96,54
GA		28	289	1	29	83,28
MOGA		8	332	1	6	95,68
MultiWavelet	207	24	236	18	29	76,8
GA		11	214	8	74	69,7
MOGA		7	257	0	43	83,72
MultiWavelet	214	9	337	6	14	92,08
GA		15	337	4	10	92,08
MOGA		3	341	4	18	93,17

Table 4.11. Simulation Results of RBBB Records Classification

Method	Record No	Normal Beats	LBBB Beats	RBBB Beats	PACED Beats	Correctly Labelled Instances %
MultiWavelet	118	1	1	333	0	99,40
GA		4	0	331	0	98,81
MOGA		2	1	332	0	99,11
MultiWavelet	124	3	4	233	17	90,66
GA		11	13	227	6	88,33
MOGA		6	10	236	5	91,83
MultiWavelet	231	3	0	243	0	98,78
GA		6	0	240	0	97,56
MOGA		1	0	245	0	99,60
MultiWavelet	212	2	1	447	0	99,33
GA		42	1	407	0	90,44
MOGA		9	0	441	0	98,00

Given a look at Table 4.12. , selection features from GA with multi-objective approach illustrates the highest accuracy results in paced records. However, Multiwavelet features are not good enough to classify pace arrhythmias according to its lowest results.

Table 4.12. Simulation Results of PACED Records Classification

Method	Record No	Normal Beats	LBBB Beats	RBBB Beats	PACED Beats	Correctly Labelled Instances %
MultiWavelet	102	8	4	0	344	96,63
GA		2	9	0	345	96,91
MOGA		0	1	0	355	99,72
MultiWavelet	104	5	7	5	467	96,49
GA		23	32	33	396	81,81
MOGA		3	1	3	477	98,55
MultiWavelet	107	0	6	0	344	98,29
GA		0	1	0	349	99,71
MOGA		0	9	0	341	97,43
MultiWavelet	217	19	32	0	304	85,63
GA		7	31	0	317	89,30
MOGA		6	38	0	311	87,61

As a result, the simulation program labelled the unlabeled signal beats as Normal, RBBB, LBBB and paced. Different sets of features of different methods have been successful in different records. Indicating that the combination of different wavelet types reflects the properties of arrhythmias which have different structures.

CHAPTER 5

ACTIVITY MONITORING WITH MOBILE ECG

Physical movements and emotional stress are the main factors having an affect on the heart rhythm and subsequently the ECG signal shape. For instance, there is a clear difference between the heart rate of sitting person and an exercising one.

In this section, a system which was designed to detect and monitor real-time heart beat of a person was explained, during the following activities; sitting, walking and running. A 3-axis accelerometer sensor determined these activities. The accelerometer data was used for motion detection. A compact casing sensor has been used to acquire and record the real-time heart signals. The real-time heart and the accelerometer data were synchronized. As a result, the heart beats can be observed and recorded during the given activities.

At first, the components that have been used in the circuit of the system will be introduced in section 5.1. And secondly, the software interface of this system will be described in section 5.2.

5.1. Hardware

5.1.1. BMD101

BMD101 is a compact component which detects bio-signals, developed by NeuroSky company for ECG applications. The internal package of component contains an analog front end (AFE) and digital filters for signal processing. Block diagram of AFE is shown in Figure 5.1. AFE block includes a high pass filter (HPF), a low noise amplifier (LNA), a sensor-off circuit and an analog digital converter (ADC).

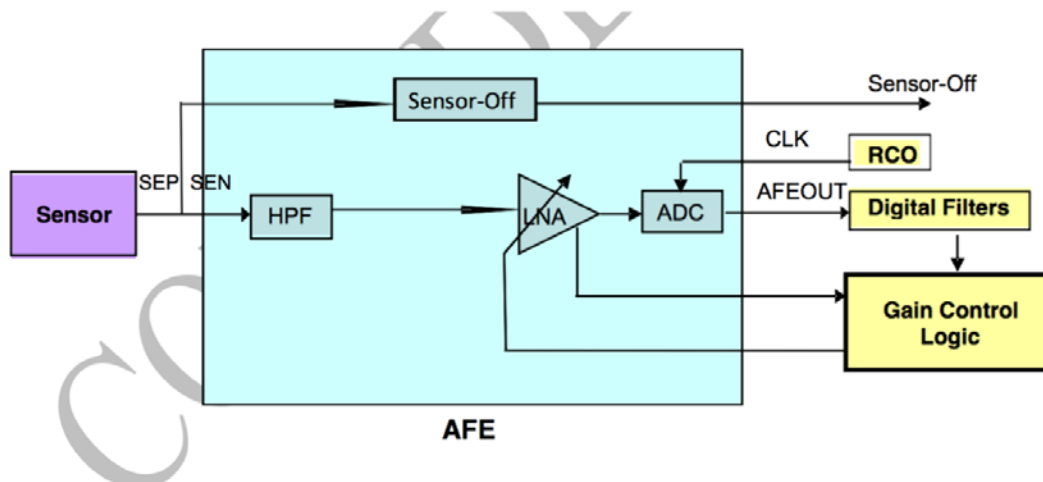


Figure 5.1. Block Diagram of AFE (From NeuroSky Inc., 2012)

The electrical activity produced from the heart is transmitted as electrical signals over the electrodes. The distances between electrodes, movement of the person or the cables connected to electrodes can put on DC components on to heart signal. HPF that integrated in BMD101 helps to remove DC components from the signal. Also the signals taken from human skin have low ranges of amplitudes such as μV . When working on these type of bio-signals, it is necessary to use the components which produce low noise. The LNA of BMD101 allows the signal to be amplified by a programmable-gain with a low system noise. The ADC used in this design has a resolution of 16-bit. The final component of the AFE is a sensor-off circuit which detects whether the signal information is coming from the electrodes or not.

The output of the ADC is fed into digital filters in order to turn the raw data into packet format. Thanks to the packet form, it is quiet easy to parse and monitor the data over an interface. The packet representation is shown in below figure.

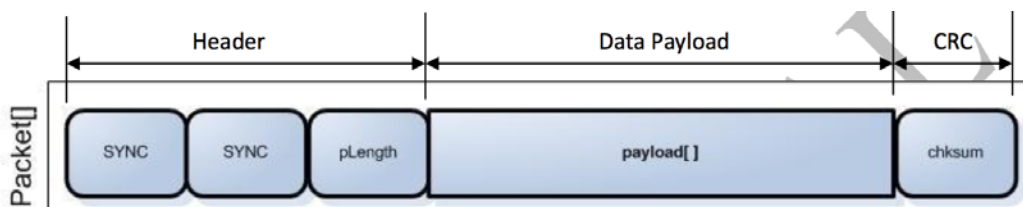


Figure 5.2. BMD101 Packet Form (From NeuroSky Inc., 2012)

In Figure 5.2 the packet includes three main parts; Header, data payload and checksum (CRC). The header has two bytes for synchronization (SYNC) and one byte for the length of the incoming payload array (pLength). Data payload contains the overall

heart signal information as an array. The plength byte illustrates the number of incoming bytes which is defined as payload[] in Figure 5.2. After getting plength byte, the transmitting bytes from BMD101 are read until the byte numbers satisfy the plength value. The next byte is checksum which must be verified to ensure that whether the packet transfer has been completed successfully or not.

BMD101 has a really tiny size and pin connection photo is shown in Figure 5.3.

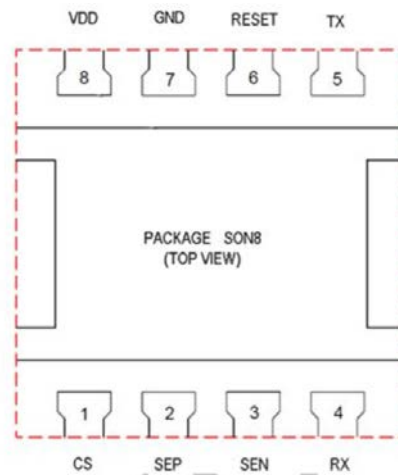


Figure 5.3. Pin Connection Diagram of BMD101 (From NeuroSky Inc., 2012)

The code segment which parses this packet will be explained in detail in software part.

5.1.2. Accelerometer

Accelerometer is a type of sensor which is generally used in measuring movements and vibrations. GY-521 is the product of Arduino and it is used as an accelerometer sensor in order to measure the movements of a person in this thesis. This microaccelerometer can sense the motions along 3-axes (x,y,z) and Figure 5.4 shows the orientation of the axes and the direction of the rotation.

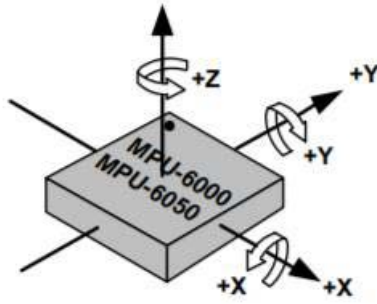


Figure 5.4. Axes and Direction Scheme of GY-521 (From InvenSense Inc. 2013)

As can be seen from Table 5.1, the accelerometer performs the measurements with respect to gravitational force (g-force). GY-521 can measure the scale ranges of $\pm 2g$, $\pm 4g$, $\pm 8g$ and $\pm 16g$. For this application device has been set to range of $\pm 2g$ by considering the accelerometer configuration register. In order to select the requested range of $\pm 2g$, Bit4 and Bit3 which are shown in Table 5.1 have been set as 0. The self-test bits of the Bit7, Bit6 and Bit5 have been also set to 0 in order to exclude them.

Table 5.1. Configuration Register Map of GY-521 (From InvenSense Inc. 2013)

Register Address (Hex)	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
1C	XA_ST	YA_ST	ZA_ST	AFS_SEL[1:0]				

Given in the accelerometer configuration register table, AFS_SEL values corresponding to the ranges are expressed in binary representation in the following Table 5.2. The sensitivity values of per LSB of accelerometer measurement output is given also in the same table under the column LSB Sensitivity.

Table 5.2. AFS_SEL Bits Configuration Values (From InvenSense Inc. 2013)

AFS_SEL	Full Scale Range	LSB Sensitivity
00	$\pm 2g$	16384 LSB/g
01	$\pm 4g$	8192 LSB/g
10	$\pm 8g$	4096 LSB/g
11	$\pm 16g$	2048 LSB/g

Accelerometer measurement registers of the product, store the measured value as 2 bytes raw data. 2 bytes raw data can be converted by using following calculation.

$$\text{Required value of axis} = \frac{\text{raw data}}{\text{LSB sensitivity for selected range}} \quad (33)$$

As a result of this calculation, the acceleration along x, y or z axes can be found in terms of g-force. The accelerometer output values x, y and z are read from the register addresses between 3B and 40 Hex. The outputs of each axes' have been stored within the registers as 16 bit raw data. These raw data are converted by using equation (33), given in microcontroller code.

5.1.3. Bluetooth Module

Bio-signals acquired from human body can be easily transferred into an interface via wireless communication modules. The most common and the cheapest way used in engineering world is Bluetooth. Fundamentally, it transmits and receives data in short distances by using serial port communication protocol.

The connection between the serial communication port of the computer and Bluetooth module is done by pairing devices with each other. In this thesis the Bluetooth module of HC-05 is used to send data from microcontroller to the computer. HC-05 includes RX and TX pins, for receiving and transmitting data respectively as can be seen in the following figure.

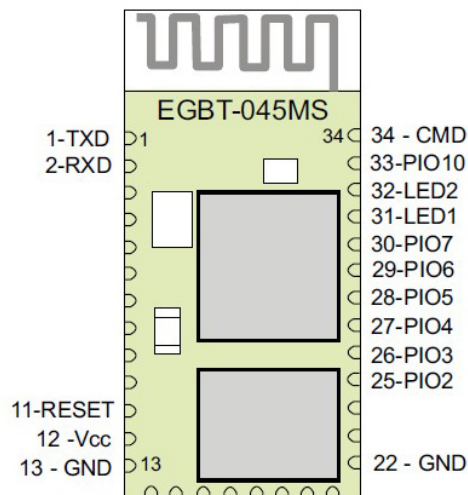


Figure 5.5. HC-05 Scheme (From Currey, 2014)

The default baud rate of BMD101 is 57600 and can not be changed. In order to transmit the data in same speed of BMD101 via Bluetooth, the baud rates should be

synchronized. The baud rate of HC-05 can be manually interfered and has been chosen for this reason in this thesis.

5.1.4. Microcontroller

Microcontroller is developed to handle specific applications such as control systems of a production line or measurement and processing of environmental temperature. Microcontrollers basically include central processing units (CPUs) with memory, input and output peripherals.

According to system design, a selected microcontroller acts as a brain. In this thesis, although the BMD101 can transmit the data directly via serial communication, the use of an accelerometer sensor entailed a microcontroller. Arduino Nano is preferred as microcontroller due to its low price and small dimensions. Both the outputs of the sensors which has been described in this chapter has been connected to Arduino Nano. A new data packet is designed to send accelerometer and BMD101 data together via bluetooth connection.

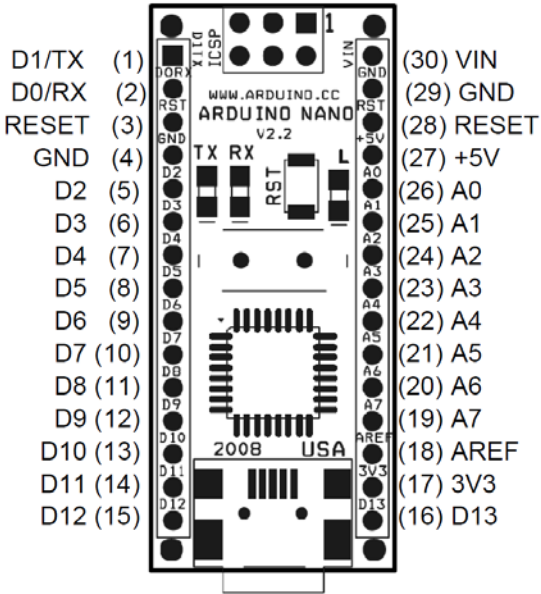


Figure 5.6. Pin Layout of Arduino Nano (From Arduino Nano V2.3 User Manuel, 2018)

Arduino Nano’s pin connection layout is shown in Figure 5.6 and description of the pins are given in Table 5.3.

Table 5.3. Pin Descriptions of Arduino Nano (From Arduino Nano V2.3 User Manuel, 2018)

Pin No.	Name	Type	Description
1-2, 5-16	D0-D13	I/O	Digital input/output port 0 to 13
3, 28	RESET	Input	Reset (active low)
4, 29	GND	PWR	Supply ground
17	3V3	Output	a+3.3V output (from FTDI)
18	AREF	Input	ADC reference
19-26	A7-A0	Input	Analog input channel 0 to 7
27	5V	Output or Input	5V output (from on-board regulator) or 5V (input from external power supply)
30	VIN	PWR	Supply voltage

5.1.5. Circuit Scheme

Circuit scheme of the system is shown in Figure 5.7. 4.5-volt chargeable battery is used with its control circuit. An on-off switch has been added to system. Since the supply voltage of bluetooth module, HC-05 is 5V and the working current value is 60mA, the supply of module is directly connected to battery. Aurduno Nano’ s 3.3V supply voltage was sufficient for the components of BMD101 and GY-521 due to the requirement of 3.3V and 30mA current.

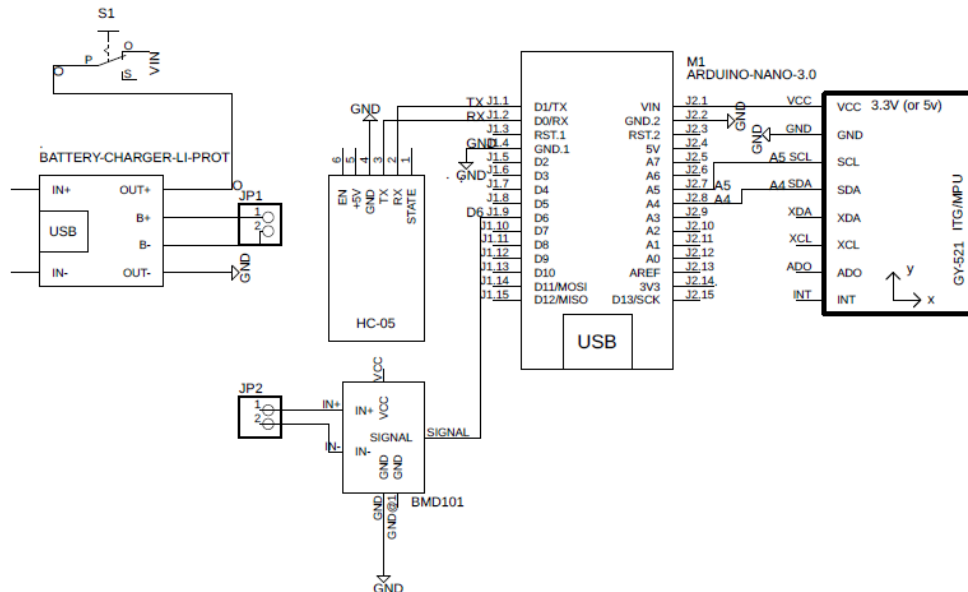


Figure 5.7. Circuit Scheme of ECG Device

The TX transfer pin of HC-05 is connected to RX receive pin of Arduino Nano and

RX receive pin of HC-05' is connected to TX transfer pin of Arduino Nano.

BMD101 has positive (+, SEP) and (-, SEN) sensor inputs to receive signals from human skin via electrodes. Instead of disposable medical electrode, a strip band used to sense the heart signals. There are two metal female sockets on the strip band and the signal is transferred to the sensor inputs SEP and SEN of BMD101 by fitting the male sockets. After the signal is processed in BMD101, it is transferred as a digital data packet to D6 pin of Arduino Nano which is the digital input. Accelerometer GY-521 is connected to Arduino Nano' s SDA (pin A5) and SCL (pin A4) in order to communicate by using I2C protocol. BMD101 and GY-521 data are processed in the microcontroller and transferred via HC-05 module' TX pin.

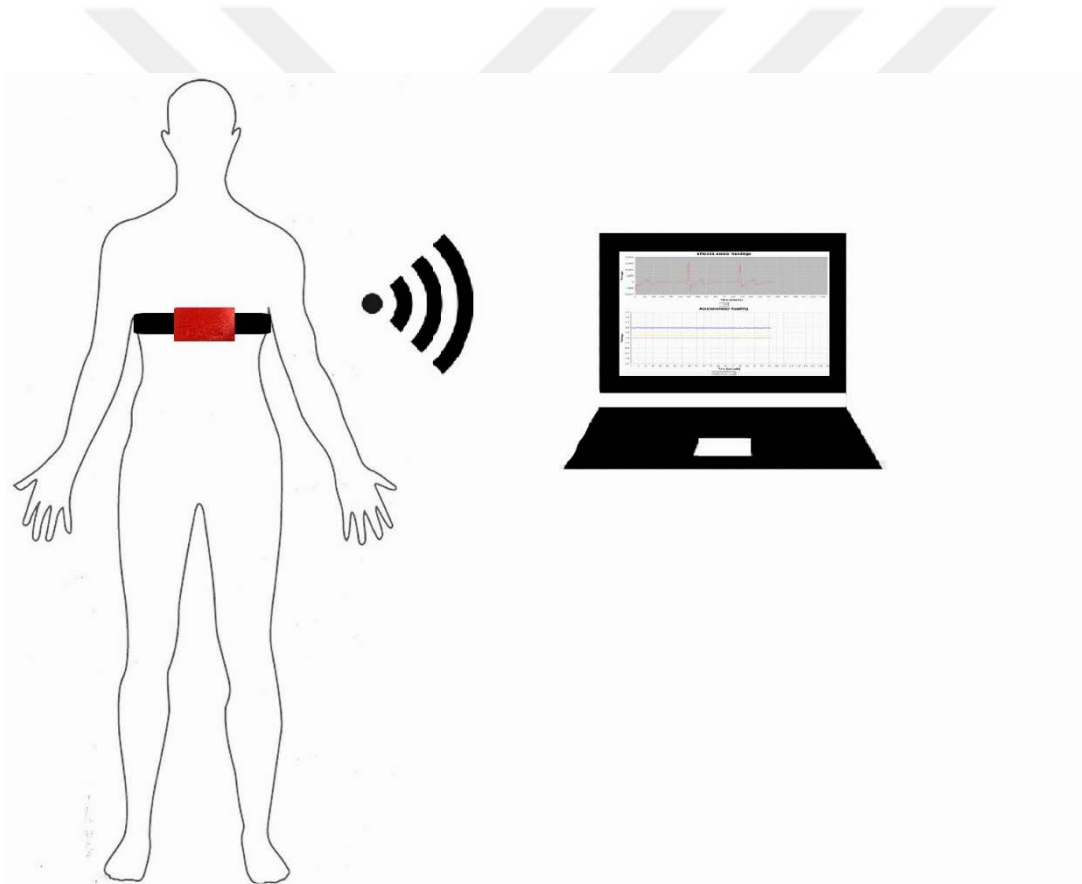


Figure 5.8. Application Schematic of Wearable ECG Device

Figure 5.8 indicates an application diagram of the system. The heart signal with accelerometer data is sending from a person to the computer with wireless. Therefore, effect of a motion on heart signal can be monitored as real-time in a software interface.

The ECG device photo is shown in Figure 5.9.



Figure 5.9. Photo of the ECG Device

5.2. Software

The data from the components in the circuit were processed in microcontroller and transferred to an interface program written in Java.

The main component of BMD101 provides heart signals by filtering. It is transmitting the data in a packet format via its TX pin (transfer port). The data can be taken from a bluetooth directly. However, hardware part includes also an accelerometer sensor GY-521. To receive and monitor both motion and ECG data together, a microcontroller was implemented to the circuit. Thus, a software which was generated in Arduino IDE (Integrated Development Environment), embedded into the microcontroller in order to obtain fast data transmission. Two functions created in code view; parsing data of BMD101 and receiving data of GY-521 from its accelerometer registers.

Parsing data of BMD101 is explained in below steps;

1. In hardware BMD101's TX pin is connected to the Arduino Nano's D6 pin. So, D6 was configured as serial pin of RX and D5 pin that has no connection, is set as TX by adding "SoftwareSerial" library of Arduino. A virtual serial port was created.
2. As long as data receives, a loop is created for continuous reading of the data from D6 (virtual RX) pin, and the incoming packet data of BMD101 is started to be examined.
3. The BMD101 packet format starts with two [SYNC] byte the hex value of 0xAA. If the first [SYNC] that caught, the algorithm continuous with next step

(NeuroSky Brain Computer Interface Technologies, 2012).

4. Next byte read from serial stream. Algorithm checks whether the byte read has a [SYNC] value. If this byte is also equal to 0xAA, the packet's initial point will be caught and program continue with the next step. If the byte is not equal to a [SYNC] byte, program returns to step 3.
5. After determining two SYNC byte, the next byte is taken as [pLength] according to output packet of BMD101. If the value of pLength is equal to 0xAA (sync byte), read the next byte until finding a byte different than 0xAA.
6. If the pLength greater than decimal value of 169, program returns to step 3. Otherwise store the value of pLength into a variable. Because the pLength byte gives the number of [DataRow] in Payload[] array.
7. Number (which is pLength content value) of bytes, were added to payload [] array and the summation result of all elements in the array is stored in a checksum accumulator.
8. Take the lowbyte of the checksum accumulator and calculate the inverse value of it.
9. Read the next byte which is [CRC] byte. If the calculation result in step 8 is equal to [CRC] byte, start the parsing process of Payload[]. Otherwise return to step 3.

Parsing a payload means separating data rows according to their missions. The Data Payload format is illustrated in Figure 5.10.

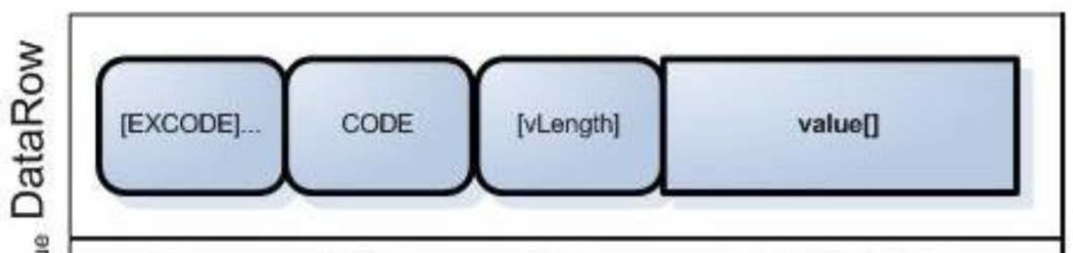


Figure 5.10. Data Row Format of the BMD101 (From NeuroSky Inc., 2012)

[EXCODE] refers to extended code level. To find the extended code level of payload array, the number of bytes which equal to 0x55 hex value are counted. The total number of [EXCODE] bytes gives the level number. [CODE] bytes have meanings according to extended code level. Each [CODE] bytes define the meaning of data in value[] array (NeuroSky Brain Computer Interface Technologies, 2012). [vLength]

gives the size of value[]. Content bytes of the payload[] were evaluated according to Table 5.4.

Table 5.4. Byte Descriptions of Payload Array (From NeuroSky Inc., 2012)

Extended Code Level	[CODE] (Hex)	[vLength] (Byte)	Data Value Meaning (value[])
0	0x02	N/A	Signal Quality (0 means sensor-off and decimal value of 200 means sensor-on)
0	0x03	N/A	Real-time Heart Rate (bpm)
0	0x08	N/A	Don't Care
0	0x80	2	16-bit Raw Data
0	0x84	5	Don't Care
0	0x85	3	Don't Care

Parsing data from Data Row Format of BMD101 explained in below steps;

1. To add data to packet that designed for transmit BMD101 data with GY-521 data to the computer software interface, a string array, which is defined as strecg, was created for this algorithm.
2. The number of [EXCODE] bytes the value of 0x55 Hex counted and stored a variable. If payload does not contain any value of 0x55 Hex, level number is set to 0.
3. If the level is 0, first element of payload array refers to [CODE] byte.
 - a) If [CODE] is equal to 0x02 then the next element of the payload gives the signal quality. Add this value to stecg by putting a label name of "POOR"
 - b) If [CODE] is equal to 0x03 then the next element of the payload gives the heart rate. Add this value to stecg by putting a label name of "HR"
 - c) If [CODE] is equal to 0x80 then the next element of the payload is vLength and it is equal to 2. After [vLength] byte, next 2 bytes give the low and high bytes of the ECG data as respectively. Add this 2 bytes value by converting byte to double and double to string to stecg by putting a front label name of "EKG". The packet content of stecg is shown in Figure 5.11.

Other [CODE] bytes and extended code levels were not considered for this application.

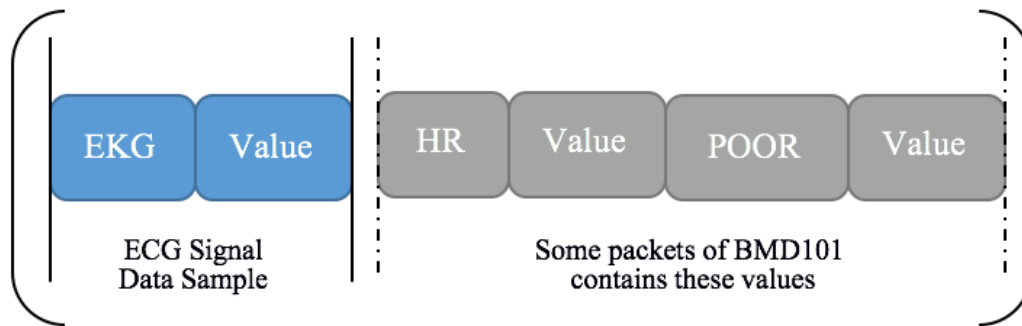


Figure 5.11. Parsed Value Packet of BMD101

The accelerometer GY-521 data was transferred to microcontroller Arduino Nano via I2C communication protocol. To constitute this communication line “Wire” library of Arduino is used.

1. Set the I2C address of the Arduino Nano to 0x68 in order to begin transmission.
2. Access the register address 0x1C which the address of accelerometer configuration register.
3. Send 0x00 hex value to the register to set the accelerometer full scale value as +/- 2g.
4. Access the accelerometer output values of `accel_x`, `accel_y` and `accel_z` registers by using the start address value. The address start with 0x3B and end with 0x40. Each address contains a byte value. Each output value (`accel_x`, `accel_y` and `accel_z`) has 2 bytes. 6 bytes were read in order starting from 0x3B address.
5. First two bytes is low and high byte of `accel_x`, second two bytes belong to `accel_y` and last two gives the `accel_z` value.
6. Each accelerometer value was divided to LSB Sensitivity value of 16384 (is given in Table 5.2).
7. Convert `accel_x`, `accel_y` and `accel_z` values to string and adding them to a string array with front labels “X”, “Y” and “Z”. String contents is shown in Figure 5.12.

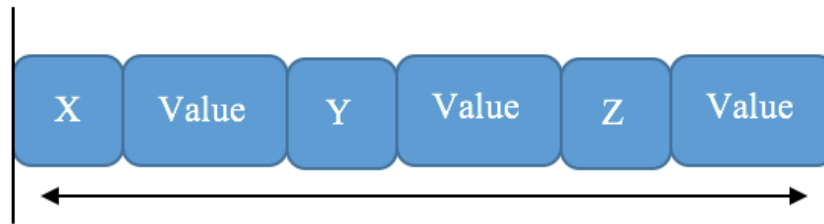


Figure 5.12. Measured Value Packet of GY521

After getting the BMD101 and GY-521 values, merge them like below figure and send the packet to the computer with serial communication protocol via Bluetooth.



Figure 5.13. Serial Data Packet of ECG Device

Software interface was created in java. Reading data from the device by using serial port of the computer. The device sends the data as given packet format in Figure 5.13. GY-521 has 1kHz frequency value and BMD101 has 512 Hz frequency. The measurement of these two components was combined into a single packet format in the form of 8 data sets of BMD101 and 1 data of GY-521 because of the frequency differences. Thus, approximately 45 data were obtained from GY-521 when 360 data were taken from BMD101 in one second. Packet is parsing according to below steps in java,

1. Search and detect the header text.
2. If string contains “EKG” text then remove the “EKG” text and get the value as string. Convert value string to double and print the value on Heart Beat Monitoring chart on interface.
3. If string contains “X”, ”Y” or ”Z” char then remove the corresponding char and get the value as string. Convert value string to double and print the value on Accelerometer chart on interface.
4. If string contains “HR” parse the value like step 2 and print it on Heart Rate label.
5. If string contains “POOR” then check the value. If value is equal to 0, then write “No connection” on screen. If value is equal to 200, then write

“Connection is ok” on screen.

5.3. Results and Discussion

The interface monitors the real-time ECG signal and motion signal at axes X, Y and Z. The screenshot views of sitting, walking and running of a healthy female person are given in below, respectively.

As expected, the heart rate value got higher as the motion got more complex, subsequently the frequency of R-peaks has increased. In Figure 5.14, since the motion is stable (sitting position), accelerometer had nothing to measure so that three constant lines have been observed. In walking activity (Figure 5.15) the R-peaks slightly became more frequent and the accelerometer has given considerable oscillations whereas the x axis of accelerometer has reached its maximum of +2g due to the defined limits in running activity (Figure 5.16). The most frequent R-peaks have been observed during running naturally.



Figure 5.14. Heart Signal and Accelerometer Signal of Sitting Person

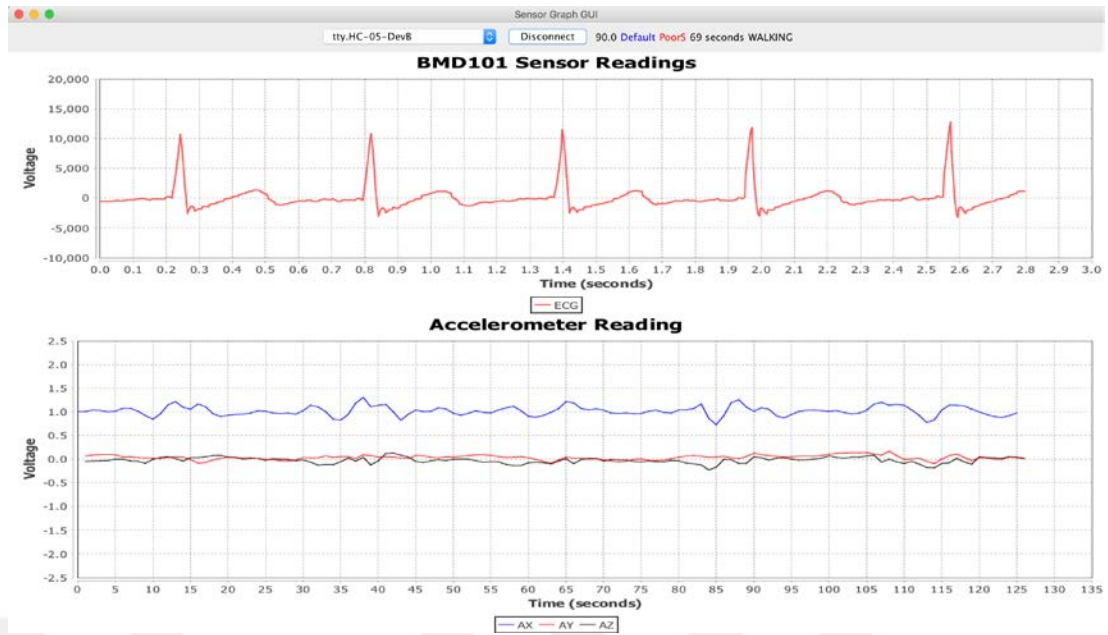


Figure 5.15. Heart Signal and Accelerometer Signal of Walking Person

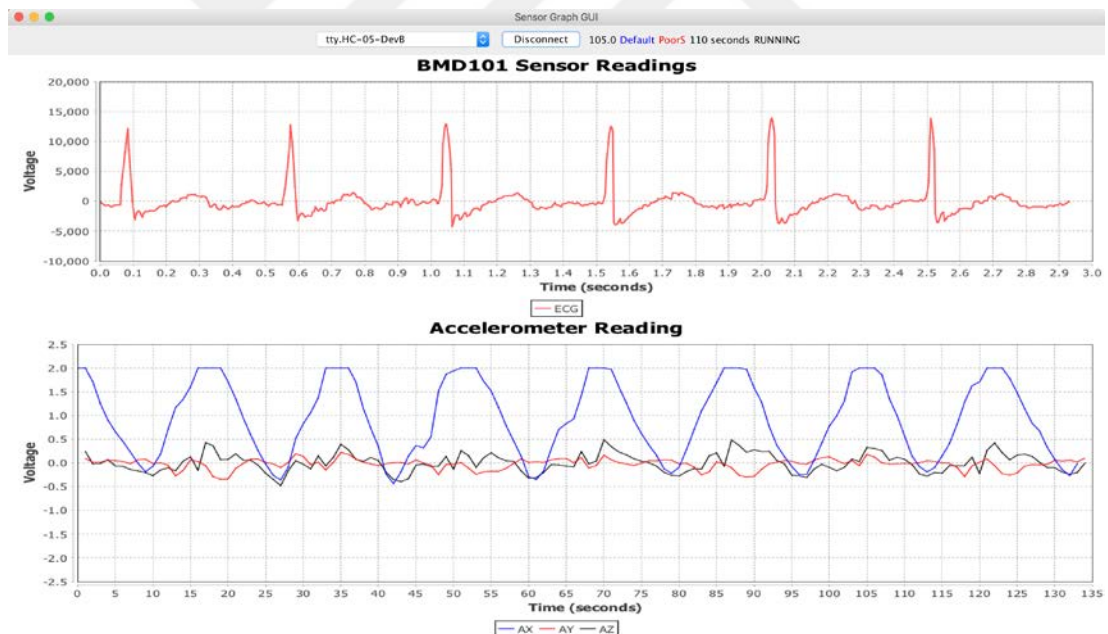


Figure 5.16. Heart Signal and Accelerometer Signal of Running Person

On GUI of the system (Figure 5.15, 5.16 and 5.17) also contains 5 indicator labels. First label (which is next to the Connect/Disconnect button) shows the real-time heart rate value. Each sample point that is taken from the device, is collecting in an array with its time stamp as milliseconds. This gives an opportunity to combine the data with its time stamp with respect to indexes. The array processed by Pan Tompkins algorithm

to examine R peak locations and the time value of the first R peak was subtracted from the time value of the last R peak in the array. Then multiplication the number of R-R peak intervals and the value 60000 (that means value of a minute in milliseconds) was divided by the result value. Figure 5.17 illustrates this HR calculation process.

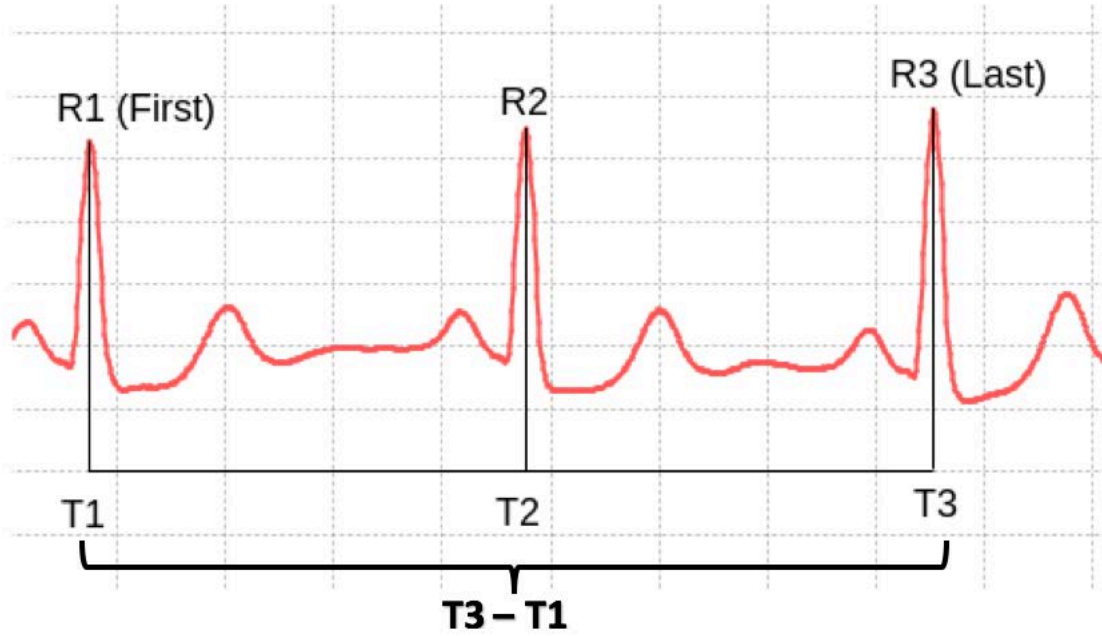


Figure 5.17. Illustration of Heart Rate Calculation

Second and third labels on GUI shows the BMD101 packet’s HR and poor signal values. These values are not included in every packet. For this reason, “Default” and “PoorS” are mentioned in these labels. If a value comes in, text of the labels changes according to the value that comes in here.

Fourth label shows the time as seconds. It indicates the time since the connection has been established and the signals have begun to be received.

The last label shows the activity. This value depends on the accelerometer x-axis data. Device is sending approximately 45 data in one second from each axis measurements. So 45 data of x-axis is collecting in an array and the highest value and the lowest value are found in this array and the difference is taken. This difference is labeled as sitting, walking and running according to the threshold values determined in the system.

The system was tested on 2 people. ECG data saved to a text file with activity labels according to accelerometer value.

First person is 30 years-old healthy female. Table 5.5 illustrates the record results of

this person. The system gives the activity labels according to accelerometer results. The number of these labels are listed under the “Activity” title of the table. The ranges of heart rate changes and average values are indicated on the table. Heart rate change was observed according to activity information.

Table 5.5. Record Results of the Online System for Female Person

Number of Beats	Seconds	Average HR	Activity	HR Range (Min-Max)
52	41,6	69	38-sitting label	63-75
53	31,9	87	27-walking label	72-92
96	51,3	99	42-running label	80-126

Second person is 32 years-old healthy male. Table 5.6 indicates the record results of this person. Number of beats, activity information and time durations were obtained from the records of system. Change of HR values due to the activity performed is clearly illustrated in Table 5.6.

Table 5.6. Record Results of the Online System for Male Person

Number of Beats	Seconds	Average HR	Activity
25	21,5	71	Sitting
71	44,2	98	Walking
67	45,1	89	Walking
56	28,85	116	Walking
35	18,4	114	Walking
49	25,7	116	Walking
50	22,2	135	Running
55	24,3	138	Running

As a result the activity information was provided in real-time. Both heart signals and the motion were observed on monitor and saved in a files.

CHAPTER 6

CONCLUSION AND FUTURE STUDIES

Analysis of ECG signals is one of the challenging subjects in biomedical signal processing world. Millions of people die every year due to heart disorders consequently early diagnosis and robust discrimination of arrhythmias have high importance. Researchers use classification methods of arrhythmias in order to detect the type of possible heart diseases. Many practical applications of biomedical signal processing can be seen in literature such as; analysis of EEG signals, investigation of muscular (EMG) signals, noise filtering in ECG signals due to sound produced in lungs. Most of these researches have been conducted by using, FT, STFT, DWT-CWT analysis techniques. Although all these methods have their advantages and disadvantages individually, in recent studies, time-frequency representations of biomedical signals are analyzed due to its' convenience of high precision diagnosis of health issues.

Two main objectives of this thesis have been carried out successfully; the first one was the analysis of cardiac arrhythmias and gathering up an efficient feature set in order to categorize heart beat anomalies. The second one was recording and monitoring of real time heartbeats by a comfortable wearable device, which indicates the type of the activity.

In digital world, it is not easy to analyze the information content of an ECG signal directly looking up on time domain representation. The electrical activity of the heart is characterized by P-QRS-T waves, which form in a cardiac cycle. By analyzing this activity, the arrhythmia shaped in a cardiac cycle can be detected. Discrimination of arrhythmias is performed by information obtained from QRS complexes. In this thesis, the QRS complex was extracted from heart signals by Pan Tompkins algorithm, which is the mostly used one in literature.

Given a look at the literature review, feature extraction from ECG signals produces data for artificial systems to construct a mathematical model of the signal. Due to the

non-stationary nature of ECG signals, statistical parameters of these signals exhibit the best characteristic information. Even better results have been obtained when the DWT detail coefficients of these signals are analyzed in same way. The ECG signal includes different frequency components in any time interval, which can be observed in detail by DWT.

In this thesis, the ECG signals downloaded from MIT-BIH database were examined by using DWT method. Four different subtypes of ECG signals, which are normal, LBBB, RBBB and paced arrhythmia beats, were downloaded from this database for improvement of ECG signal classification. In order to prepare the data to be ready to analyze fairly, each fragment's QRS complexes' were detected and extracted by using Pan Tompkins algorithm.

There are many approaches for classification algorithms. The most widespread one for bio-signals which is neural networks were used in this study. It is important to select the appropriate training function. To determine a suitable training function for ECG data, features were extracted from DB4 wavelet, which exhibits a matching characteristic to the ECG signal shape, and all training functions in Matlab NN Toolbox were tested according to these features.

As a first step, QRS complexes of corresponding beats were transformed with the DB4 wavelet and detail coefficients were obtained for 4 levels. Statistical parameters which are mean, standard deviation, energy and entropy, were calculated using each level's detail coefficients to create features of the heart beats. 11 different types of training function were tested in neural network and the accuracy results have been compared. The training function, which gave the best result with accuracy rate of 92.27 %, was Levenberg-Maduquart (trainlm). After determining the best training function, features were extracted by using 13 different wavelet types. Mean, standard-deviation, energy and entropy features have been computed for each level, 4 levels in the end. Therefore, one wavelet provided 16 features by generating 4 level detail coefficients. These features were tested in neural network's Levenberg-Maduquart training function to find the best wavelet type that provides highest discriminative property when these features are in use. DB4 wavelet shows the highest performance by providing 95,09 % accuracy when the accuracy rate of the train data have been compared.

However, when the statistical values of the classification algorithm were examined

separately for 4 different arrhythmia types, it was observed that different wavelets showed high performance for different arrhythmia types. Therefore, the feature set obtained by the combinations of the best two wavelets for arrhythmia types was selected and tested in NN classification algorithm. DB1 provides the greatest average accuracy of 97.02 % for all classes. For RBBB rhythm DB3 gives 98,20 % and SYM4 wavelet provides 98,18 % accuracy rate, SYM3's accuracy rate is 97,83 % for LBBB rhythm and DB4 for paced arrhythmia gives 97,38 %. Thus, 32 features extracted from the combination of DB1 with DB3, DB4, SYM3 and SYM4 were tested in neural network classification. The combination of DB1 and DB4 gives the best accuracy rate of 98,92 % when compared with the 16 feature set obtained from a single wavelet. Higher classification performance was achieved by combining the features obtained from the 4-level detail coefficients of different wavelet types.

After these results, the effects of combination of different wavelet types for various level detail coefficients were investigated by GA method. Extracting 208 features from 13 different wavelets by using 4-level detail coefficients, a feature pool has been constructed. By using GA, feature sets were searched in this pool to find better features which might have higher discriminative properties for these 4 arrhythmia types. Fitness function of GA has been set as accuracy rate of MLP NN. The fitness function has been tried to be optimized by randomly selecting detail coefficients of any wavelet in any level in the pool.

As a result of combination different wavelet's different level detail coefficients, 16 features were found by GA which gives the classification result of 97,36 % of fitness function value, with 100 generations and 35 population number. It has been observed that high performance features can be selected using a GA.

In feature selection of GAs generally controls one parameter in fitness assignment. Sometimes this point of view restricts the examination of different structures of the data that used. The neural networks used in the classification of ECG signals, calculate and update the weights depending on the different components of the data in the training process. Looking at the different evaluation results of the neural network in order to increase the performance ratio of selected features, enables more efficient solutions in this process. Therefore, GA was designed by using multi-objective approach to add different parameters. In multi-objective approach fitness function constituted by using three criteria; RMSE of neural network, number of selected

features and accuracy rate of the neural network. The main idea is to reduce the classification error rate and obtain high accuracy with minimum number of features. 48 features selected by using GA with multi-objective approach. These features provide 98,79 % accuracy rate in MLP NN.

Three different methods were used to find for features that provided a high success rate in classification of ECG arrhythmias. The simulation software created by using these three method feature sets; Multiwavelet features, feature set that found by GA and feature set that found by GA with multi-objective approach. Each type of ECG signal separated with a ratio of 80 % train data and 20 % testing. Trained MLP results implemented into the software and using it, unlabeled beats of simulation signal beats labelled. Comparing these three method's features sets, GA with multi-objective approach gives the highest accuracy rate value which is 96.69 %.

In this thesis also a compact wearable ECG device was designed for examining activity-related heartbeats. A software interface designed to monitor accelerometer and ECG sensor values. It was observed that the heart rate changed during physical activity and this change matched the knowledge of the accelerometer sensor. Records were obtained by designed ECG device that monitors motion information and real-time heart signal.

In future, we aim to combine the two objectives of this thesis, simply realize arrhythmia detection in our hardware design, possibly with a mobile application based on a cloud service. In addition, number of the activities sensed by the device also can be increased. Other than DWT, there are far more accepted ECG analysis methods in literature, these various investigation approaches might be used to obtain greater accuracy results.

REFERENCES

- Addison, P. S. (2017). *The illustrated wavelet transform handbook: introductory theory and applications in science, engineering, medicine and finance*. CRC press.
- Arduino Nano V2.3 User Manuel,
<https://www.arduino.cc/en/uploads/Main/ArduinoNanoManual23.pdf> (Access Date: 01 June 2018)
- Auger, F., Flandrin, P., Gonçalvès, P., & Lemoine, O. (1996). Time-frequency toolbox. CNRS France-Rice University, 46.
- Bailey, R. (2018), “Diastole and Systole Phases of the Cardiac Cycle”,
<https://www.thoughtco.com/phases-of-the-cardiac-cycle-anatomy-373240>
(Access Date: 14 August 2018)
- Beyramienanlou, H., & Lotfivand, N. (2017). Shannon’s energy based algorithm in ECG signal processing. Computational and mathematical methods in medicine, 2017.
- Castro, B., Kogan, D., & Geva, A. B. (2000). ECG feature extraction using optimal mother wavelet. In Electrical and electronic engineers in israel, 2000. the 21st ieee convention of the (pp. 346-350). IEEE.
- Cîmpanu, C., Ferariu, L., Dumitriu, T., & Ungureanu, F. (2017, June). Multi-Objective Optimization of Feature Selection procedure for EEG signals classification. In E-Health and Bioengineering Conference (EHB), 2017 (pp. 434-437). IEEE.
- Currey, M. (2014), “HC-05 and HC-06 zs-040 Bluetooth modules. First Look”,
<http://www.martyncurrey.com/hc-05-and-hc-06-zs-040-bluetooth-modules-first-look/> (Access Date: July 2018)
- de Albuquerque, V. H. C., Nunes, T. M., Pereira, D. R., Luz, E. J. D. S., Menotti, D., Papa, J. P., & Tavares, J. M. R. (2018). Robust automated cardiac arrhythmia detection in ECG beat signals. *Neural Computing and Applications*, 29(3), 679-693.
- De Chazal, P., O’Dwyer, M., & Reilly, R. B. (2004). Automatic classification of heartbeats using ECG morphology and heartbeat interval features. *IEEE transactions on biomedical engineering*, 51(7), 1196-1206.
- Engin, M., Fedakar, M., Engin, E. Z., & Korürek, M. (2007). Feature measurements of ECG beats based on statistical classifiers. *Measurement*, 40(9-10), 904-912.
- Fonseca, C. M., & Fleming, P. J. (1993, June). Genetic Algorithms for Multiobjective Optimization: Formulation Discussion and Generalization. In *Icga* (Vol. 93, No. July, pp. 416-423).

- Gonçalves, I. B., Leiria, A., & Moura, M. M. M. (2013). STFT or CWT for the detection of Doppler ultrasound embolic signals. *International journal for numerical methods in biomedical engineering*, 29(9), 964-976.
- Haykin, S. S., Haykin, S. S., Haykin, S. S., & Haykin, S. S. (2009). *Neural networks and learning machines* (Vol. 3). Upper Saddle River, NJ, USA:: Pearson.
- Huang, C. L., & Wang, C. J. (2006). A GA-based feature selection and parameters optimization for support vector machines. *Expert Systems with applications*, 31(2), 231-240.
- Hussain, S. A., Singh, A. V., Ramaiah, C. S., & Hussain, S. J. (2016, September). A wireless device for patient ECG monitoring and motion activity recording for medical applications. In *Reliability, Infocom Technologies and Optimization (Trends and Future Directions)(ICRITO), 2016 5th International Conference on* (pp. 634-641). IEEE.
- Ince, T., Kiranyaz, S., & Gabbouj, M. (2009). A generic and robust system for automated patient-specific classification of ECG signals. *IEEE Transactions on Biomedical Engineering*, 56(5), 1415-1426.
- Patro, K. K., & Kumar, P. R. (2017). Effective Feature Extraction of ECG for Biometric Application. *Procedia Computer Science*, 115, 296-306.
- Yu, S. N., & Chou, K. T. (2008). Integration of independent component analysis and neural networks for ECG beat classification. *Expert Systems with Applications*, 34(4), 2841-2846.
- InvenSense Inc. (2013), MPU-6000/MPU-6050 Product Specification Revision 3.4, <https://www.invensense.com/wp-content/uploads/2015/.../MPU-6000-Datasheet1.pdf> (Access Date: 01 June 2018)
- InvenSense Inc. (2013), MPU-6000/MPU-6050 Register Map and Descriptions Revision 4.2, <https://www.invensense.com/wp-content/uploads/.../MPU-6000-Register-Map1.pdf> (Access Date: 01 June 2018)
- Jiang, X., Zhang, L., Zhao, Q., & Albayrak, S. (2006, November). ECG arrhythmias recognition system based on independent component analysis feature extraction. In *TENCON 2006. 2006 IEEE Region 10 Conference* (pp. 1-4). IEEE.
- Kim, B. H., Noh, Y. H., & Jeong, D. U. (2015, March). A wearable ECG monitoring system using adaptive EMD filter based on activity status. In *Advanced Information Networking and Applications Workshops (WAINA), 2015 IEEE 29th International Conference on* (pp. 11-16). IEEE.
- Leardi, R., & Gonzalez, A. L. (1998). Genetic algorithms applied to feature selection in PLS regression: how and when to use them. *Chemometrics and intelligent laboratory systems*, 41(2), 195-207.
- Machine Learning Group at the University of Waikato, "Weka 3: Data Mining Software in Java", <https://www.cs.waikato.ac.nz/ml/weka/> (Access Date: January

2018)

- Mallat, S. G. (1989). A theory for multiresolution signal decomposition: the wavelet representation. *IEEE transactions on pattern analysis and machine intelligence*, 11(7), 674-693.
- Mazomenos, E. B., Chen, T., Acharyya, A., Bhattacharya, A., Rosengarten, J., & Maharatna, K. (2012, March). A time-domain morphology and gradient based algorithm for ECG feature extraction. In *Industrial Technology (ICIT), 2012 IEEE International Conference on* (pp. 117-122). IEEE.
- Mironovova, M., & Bíla, J. (2015, July). Fast fourier transform for feature extraction and neural network for classification of electrocardiogram signals. In *Future Generation Communication Technology (FGCT), 2015 Fourth International Conference on* (pp. 1-6). IEEE.
- MIT-BIH Arrhythmia Database Directory,
<https://www.physionet.org/physiobank/database/html/mitdbdir/mitdbdir.htm>
(Access Date: 2016)
- Moody, G. B., & Mark, R. G. (2001). The impact of the MIT-BIH arrhythmia database. *IEEE Engineering in Medicine and Biology Magazine*, 20(3), 45-50.
- NeuroSky Brain Computer Interface Technologies (2012, March). BMD100 Communications Protocol.
- NeuroSky Inc. (2012, June). BMD101 Product Brief (Preliminary).
- Oreski, S., & Oreski, G. (2014). Genetic algorithm-based heuristic for feature selection in credit risk assessment. *Expert systems with applications*, 41(4), 2052-2064.
- Pan, J., & Tompkins, W. J. (1985). A real-time QRS detection algorithm. *IEEE Trans. Biomed. Eng.*, 32(3), 230-236.
- Patro, K. K., & Kumar, P. R. (2017). Effective Feature Extraction of ECG for Biometric Application. *Procedia Computer Science*, 115, 296-306.
- Pławiak, P. (2018). Novel methodology of cardiac health recognition based on ECG signals and evolutionary-neural system. *Expert Systems with Applications*, 92, 334-349.
- Rangayyan, R. M. (2015). *Biomedical signal analysis* (Vol. 33). John Wiley & Sons.
- Sannino, G., & De Pietro, G. (2018). A deep learning approach for ECG-based heartbeat classification for arrhythmia detection. *Future Generation Computer Systems*.
- Sarkaleh, M. K., & Shahbahrami, A. (2012). Classification of ECG arrhythmias using discrete wavelet transform and neural networks. *International Journal of Computer Science, Engineering and Applications*, 2(1), 1.

- Sarvan, Ç., & Özkurt, N. (2017, May). Multiwavelet feature sets for ECG beat classification. In *Signal Processing and Communications Applications Conference (SIU), 2017 25th*(pp. 1-4). IEEE.
- Sarvan, Ç., & Özkurt, N. (2018, May). Feature selection for ECG beat classification using genetic algorithms with a multi-objective approach. In *2018 26th Signal Processing and Communications Applications Conference (SIU)*. IEEE.
- Sarvan, Ç., Özkurt, N., Karabulut, K. (2017, November). Feature Selection for ECG Beat Classification using Genetic Algorithms. *Akıllı Sistemlerde Yenilikler ve Uygulamaları (ASYU), 2017* .
- Shufni, S. A., & Mashor, M. Y. (2015, March). ECG signals classification based on discrete wavelet transform, time domain and frequency domain features. In *Biomedical Engineering (ICoBE), 2015 2nd International Conference on*(pp. 1-6). IEEE.
- Sun, F., Yi, C., Li, W., & Li, Y. (2017). A wearable H-shirt for exercise ECG monitoring and individual lactate threshold computing. *Computers in Industry*, 92, 1-11.
- Talbi, E. G. (2009). *Metaheuristics: from design to implementation* (Vol. 74). John Wiley & Sons.
- Tripathy, R. K., Mendez, A. Z., de la O, S., Arrieta Paternina, M. R., Arrieta, J. G., & Naik, G. R. (2018). Detection of Life Threatening Ventricular Arrhythmia using Digital Taylor Fourier Transform. *Frontiers in Physiology*, 9, 722.
- Uslu, E., & Bilgin, G. (2008, April). Classification of heart arrhythmias by using wavelet and merged wavelet packet transforms. In *Signal Processing, Communication and Applications Conference, 2008. SIU 2008*. IEEE 16th (pp. 1-4). IEEE.
- Watchmaker Framework for Evolutionary Computation,
<https://watchmaker.uncommons.org/> (Access Date: May 2017)
- Wikipedia, the free encyclopedia, “QRS Complex”,
https://en.wikipedia.org/wiki/QRS_complex (Access Date: 2016)
- Xia, H., Asif, I., & Zhao, X. (2013). Cloud-ECG for real time ECG monitoring and analysis. *Computer methods and programs in biomedicine*, 110(3), 253-259.
- Yang, J., & Honavar, V. (1998). Feature subset selection using a genetic algorithm. In *Feature extraction, construction and selection* (pp. 117-136). Springer, Boston, MA.
- Yu, S. N., & Chou, K. T. (2008). Integration of independent component analysis and neural networks for ECG beat classification. *Expert Systems with Applications*, 34(4), 2841-2846.

APPENDIX 1 –Decomposition Filter Coefficients of Wavelet

The high pass and low pass filter coefficients obtained from Matlab. In simulation program of java below values were used.

DAUBECHIES1

LPF=[7.071068E-01;7.071068E-01;]
HPF =[-7.071068E-01;7.071068E-01;]

DABUCHIES2

LPF=[-1.294095E-01;2.241439E-01;8.365163E-01;4.829629E-01;]
HPF=[-4.829629E-01;8.365163E-01;-2.241439E-01;-1.294095E-01;]

DAUBECHIES4

LPF=[-1.059740E-02;3.288301E-02;3.084138E-02;-1.870348E-01;-2.798377E-02;6.308808E-01;7.148466E-01;2.303778E-01;]
HPF=[-2.303778E-01;7.148466E-01;-6.308808E-01;-2.798377E-02;1.870348E-01;3.084138E-02;-3.288301E-02;-1.059740E-02;]

DAUBECHIES6

LPF=[-1.077301E-03;4.777258E-03;5.538422E-04;-3.158204E-02;2.752287E-02;9.750161E-02;-1.297669E-01;-2.262647E-01;3.152504E-01;7.511339E-01;4.946239E-01;1.115407E-01;]
HPF=[-1.115407E-01;4.946239E-01;-7.511339E-01;3.152504E-01;2.262647E-01;-1.297669E-01;-9.750161E-02;2.752287E-02;3.158204E-02;5.538422E-04;-4.777258E-03;-1.077301E-03;]

DAUBECHIES10

LPF=[-1.326420E-05;9.358867E-05;-1.164669E-04;-6.858567E-04;1.992405E-03;1.395352E-03;-1.073318E-02;3.606554E-03;3.321267E-02;-2.945754E-02;-7.139415E-02;9.305736E-02;1.273693E-01;-1.959463E-01;-2.498464E-01;2.811723E-01;6.884590E-01;5.272012E-01;1.881768E-01;2.667006E-02;]
HPF=[-2.667006E-02;1.881768E-01;-5.272012E-01;6.884590E-01;-2.811723E-01;-2.498464E-01;1.959463E-01;1.273693E-01;-9.305736E-02;-7.139415E-02;2.945754E-02;3.321267E-02;-3.606554E-03;-1.073318E-02;-1.395352E-03;1.992405E-03;6.858567E-04;-1.164669E-04;-9.358867E-05;-1.326420E-05;]

SYMLET6

LPF=[1.540411E-02;3.490712E-03;-1.179901E-01;-4.831174E-02;4.910559E-01;7.876411E-01;3.379294E-01;-7.263752E-02;-2.106029E-02;4.472490E-02;1.767712E-03;-7.800708E-03;]
HPF=[7.800708E-03;1.767712E-03;-4.472490E-02;-2.106029E-02;7.263752E-02;3.379294E-01;-7.876411E-01;4.910559E-01;4.831174E-02;-1.179901E-01;-

3.490712E-03;1.540411E-02;]

COIFLET5

LPF=[-9.517657E-08;-1.674429E-07;2.063762E-06;3.734655E-06;-2.131503E-05;-
4.134043E-05;1.405411E-04;3.022596E-04;-6.381313E-04;-1.662864E-
03;2.433373E-03;6.764185E-03;-9.164231E-03;-1.976178E-02;3.268357E-
02;4.128921E-02;-1.055742E-01;-6.203596E-02;4.379916E-01;7.742896E-
01;4.215662E-01;-5.204316E-02;-9.192001E-02;2.816803E-02;2.340816E-02;-
1.013112E-02;-4.159359E-03;2.178236E-03;3.585897E-04;-2.120808E-04;]
HPF=[2.120808E-04;3.585897E-04;-2.178236E-03;-4.159359E-03;1.013112E-
02;2.340816E-02;-2.816803E-02;-9.192001E-02;5.204316E-02;4.215662E-01;-
7.742896E-01;4.379916E-01;6.203596E-02;-1.055742E-01;-4.128921E-
02;3.268357E-02;1.976178E-02;-9.164231E-03;-6.764185E-03;2.433373E-
03;1.662864E-03;-6.381313E-04;-3.022596E-04;1.405411E-04;4.134043E-05;-
2.131503E-05;-3.734655E-06;2.063762E-06;1.674429E-07;-9.517657E-08;]

BIORTHOGONAL1.5

LPF=[1.657282E-02;-1.657282E-02;-1.215340E-01;1.215340E-01;7.071068E-
01;7.071068E-01;1.215340E-01;-1.215340E-01;-1.657282E-02;1.657282E-02;]
HPF=[0;0;0;0;-7.071068E-01;7.071068E-01;0;0;0;0;]

BIORTHOGONAL2.6

LPF=[0;-6.905340E-03;1.381068E-02;4.695631E-02;-1.077233E-01;-1.698714E-
01;4.474660E-01;9.667476E-01;4.474660E-01;-1.698714E-01;-1.077233E-
01;4.695631E-02;1.381068E-02;-6.905340E-03;]
HPF=[0;0;0;0;0;3.535534E-01;-7.071068E-01;3.535534E-01;0;0;0;0;0;]

BIORTHOGONAL3.5

LPF=[-1.381068E-02;4.143204E-02;5.248058E-02;-2.679272E-01;-7.181553E-
02;9.667476E-01;9.667476E-01;-7.181553E-02;-2.679272E-01;5.248058E-
02;4.143204E-02;-1.381068E-02;]
HPF=[0;0;0;0;-1.767767E-01;5.303301E-01;-5.303301E-01;1.767767E-01;0;0;0;0;]

BIORTHOGONAL5.5

LPF=[0;0;3.968709E-02;7.948109E-03;-5.446379E-02;3.456053E-01;7.366602E-
01;3.456053E-01;-5.446379E-02;7.948109E-03;3.968709E-02;0;]
HPF=[-1.345671E-02;-2.694967E-03;1.367066E-01;-9.350470E-02;-4.768033E-
01;8.995061E-01;-4.768033E-01;-9.350470E-02;1.367066E-01;-2.694967E-03;-
1.345671E-02;0;]