# ECG DATA COMPRESSION USING ARTIFICIAL NEURAL NETWORKS

**A DISSERTATION**

***Submitted in partial fulfillment of the requirements for the award of the degree of***

**MASTER OF TECHNOLOGY**

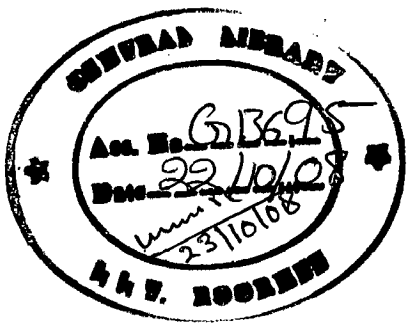
**in**

**ELECTRICAL ENGINEERING**

**(With Specialization in Measurement and Instrumentation)**

**By**

**FASIL KIROS ALEMAYEHU**





**DEPARTMENT OF ELECTRICAL ENGINEERING**
**INDIAN INSTITUTE OF TECHNOLOGY ROORKEE**
**ROORKEE - 247 667 (INDIA)**
**JUNE, 2008**

# CANDIDATE'S DECLARATION

I hereby certify that the work which is being presented in the dissertation with a title **'ECG DATA COMPRESSION USING ARTIFICIAL NEURAL NETWORKS'** in partial fulfillment of the requirements for the award of the degree of Master Of Technology in Electrical Engineering with specialization in Measurements and Instrumentation, submitted to the department of Electrical Engineering, Indian Institute of Technology Roorkee, India is an authentic record of my own work carried out during a period from June 2007 to June 2008 under the supervision of **Dr. VINOD KUMAR**, Professor, Electrical Engineering department, Indian Institute of Technology, Roorkee and **Ms. AMBALIKA SHARMA,** Assistant Professor, Electrical Engineering department, Indian Institute of Technology, Roorkee.

The matter presented in this dissertation has not been submitted by me for the award of any other degree of this Institute or any other Institute

**(FASIL KIROS ALEMAYEHU)**

Dated: June 26, 2008

## CERTIFICATE

This is to certify that the above statement made by the candidate is correct to the best of my knowledge and belief.

26/6

**(Dr. VINOD KUMAR)**
Professor
Department of Electrical Engineering
IIT Roorkee,
Roorkee-247667 (India)

Ambalika Sharma

**(Ms. AMBALIKA SHARMA)**
Assistant Professor 26/6/08
Department of Electrical Engineering
IIT Roorkee,
Roorkee-247667 (India)

# ACKNOWLEDGEMENT

First of all I would like to thank **God** for helping me reach this moment.

Then I would like to give my sincere gratitude to **Dr. Vinod Kumar, professor, Electrical Engineering Department, Indian Institute of Technology Roorkee, Roorkee**, for guiding me through this dissertation work with the topic **"ECG DATA COMPRESSION USING ARTIFICIAL NEURAL NETWORKS"**, as well as for providing me all the necessary materials, for his value adding knowledge, for directing me through the right paths of the work, and the inspirational support throughout this dissertation.

I want to express my earnest gratefulness to **Ms. Ambalika Sharma, assistant professor, Electrical Engineering Department, Indian Institute of Technology Roorkee, Roorkee**, for her guidance, for giving me helpful materials and encouraging me to succeed in my work.

My heartfelt gratitude goes to all the teachers of Measurements and Instrumentation group who, with their encouraging and caring words, constructive criticism and suggestions have contributed directly or indirectly in a significant way towards completion of this dissertation.

I convey my deep sense of gratitude to the Head of Electrical Engineering Department, who directly or indirectly helped me during the work.

**FASIL KIROS ALEMAYEHU**

M.Tech (Measurement & Instrumentation)

# ABSTRACT

Multi-channel electrocardiograms (ECGs) of high precision are the base and modern techniques for the non-invasive monitoring of cardiac conditions. The quality and accuracy of medical diagnosis is increasing due to improved probing techniques and instrumentation. At the same time, it also implies that vast amounts of data are generated. Usually the electrocardiogram needs to be stored as part of the clinical record or for further diagnosis or to be transmitted for real-time tele-diagnosis and monitoring.

A recommended remedy of solving the costly economy of storing and transmitting these signals is to compress them. Even though Many ECG compression techniques have been developed before, most of them do not meet the requirements for clinical acceptability.

This work intends to develop a compression system by adopting the application of Artificial Neural Network in dimension reduction. This system first pre-processes the ECG data using digital filters to improve the quality of the ECG signals. Then it adopts amplitude and first derivative algorithm to detect the R-points accurately. Finally the partitioned beats are trained using artificial neural network which adopts a robust learning algorithm called Resilient Backpropagation algorithm. Weights and hidden layer activations are stored to represent the original ECG data. Impermanent and highly aberrant ECG beats are stored uncompressed.

The outcome of this result shows, using backpropagation artificial neural network based compression system is an efficient and effective way of compression ECG signals with high compression ratio, improved compression precision and less compression time. This technique also provides high reconstruction fidelity.

# TABLE OF CONTENTS

Page No.

# LIST OF FIGURES

Page

# LIST OF TABLES

**Page**

# CHAPTER ONE

# INTRODUCTION

## 1.1 Background and motivation

Multi-channel electrocardiograms (ECGs) of high precision are the base and modern techniques for the non-invasive monitoring of cardiac conditions. This improvement in probing techniques and instrumentation provides more accurate information for medical diagnosis. At the same time, it also implies that vast amounts of data are generated. Usually the electrocardiogram needs to be stored as part of the clinical record or for further diagnosis or to be transmitted for real-time tele-diagnosis and monitoring.

It can be argued that the continuous improvement in probing equipment which generates increasingly higher data rates should not be hindered by the limits of storage systems and communication infrastructure. Similarly the economy of storing the data is often just as important as the possibility of collecting the data.

This has motivated the search for advanced data compression techniques specifically designed for multi-channel ECG data. Different compression techniques have already been established and each has a different level of clinical acceptability.

It is well known that artificial neural networks can be applied to dimensionality reduction and such manipulation of data is a common preprocessing technique for some pattern recognition and classification applications. This dissertation suggests and explores the application of artificial neural networks to solve the ECG data compression problem.

If one defines to size of data as its dimensionality reducing the dimensionality of the data achieves compression.

The application of neural network techniques to data compression is relatively new and therefore this dissertation could serve as a contribution for further work. It seeks to identify the most promising directions for achieving real-time high compression ratios and hence low-loss ECG data compression.

By using the optimal methods of data processing and network training developed during the Dissertation work we aim to outperform traditional methods in both compression ratio and error performance.

## 1.2 Organization of the Report

- **Chapter 2** discusses what the ECG signal is, where it originates from and some mechanical aspects of it. It also discusses the basic components of the ECG signal and describes the features of these components during normal and abnormal conditions. Finally it explains about the ECG leads and their configurations and declares which kind of configuration is selected for this work.
- The theoretical aspects of Artificial Neural Networks (ANN), their historical background, neuron models, transfer functions and network architectures is discussed in detail in **Chapter 3**. It also explains about feedforward networks and the backpropagation algorithm.
- **Chapter 4** gives a description of the existing multi-channel ECG compression techniques, explains how each technique works and gives a comparison of each technique with one another.

- **Chapter 5** is devoted to the proposed technique for ECG data compression. It explains how ANNs can be used for ECG data compression and provides a detail on what procedures are followed, which BP algorithm is selected and how the performance of the proposed compression system is evaluated. It also discusses how the compressed ECG data are reconstructed.
- Based on the techniques and procedures discussed in **Chapter 5**, the results of the performed experiments are presented **in Chapter 6**. This chapter also presents a comparison of the proposed technique with the existing ones based on the results obtained.
- **Chapter 7** summarizes the key results and observations to produce an overall conclusion. This chapter also recommends directions for future work.

# CHAPTER TWO

# THE ELECTROCARDIOGRAM (ECG)

## 2.1 Introduction

Before developing a compression system using backpropagation artificial neural networks or, in general, any signal processing technique for a particular signal, it is necessary to have a thorough knowledge about what kind of signal it is, how it is formed and what its characteristics are, and what the building blocks of the signal are.

This chapter discusses what type of signal the ECG signal is and where it originates from. It describes the mechanical aspects of the ECG signal by showing the mechanical and electrical events during a single cardiac cycle.

The basic components of the ECG signal and their characteristics during normal and abnormal conditions are given a section in this chapter. This section serves as a reference during comparison of the network trained ECGs and the original ECGs. Studying this section helps to identify whether the cause of aberrations in the reconstructed ECG data is a "real" abnormality or an artifact caused by the compression system.

Finally the chapter explains about the ECG leads and their configurations and declares which kind of configuration is selected for this work.

## 2.2 The origin of the ECG

The biopotentials generated by the muscles of the heart result in the electrocardiogram (ECG). Each action potential in the heart originates near the top of the right atrium of the heart, at a point called the sinoatrial (SA) node [1] (see figure2.1).

The SA node is a group of specialized cells that spontaneously generate action potentials at a regular rate that propagate in all directions along the surface of the atria and finally reach the surface of the ventricles.

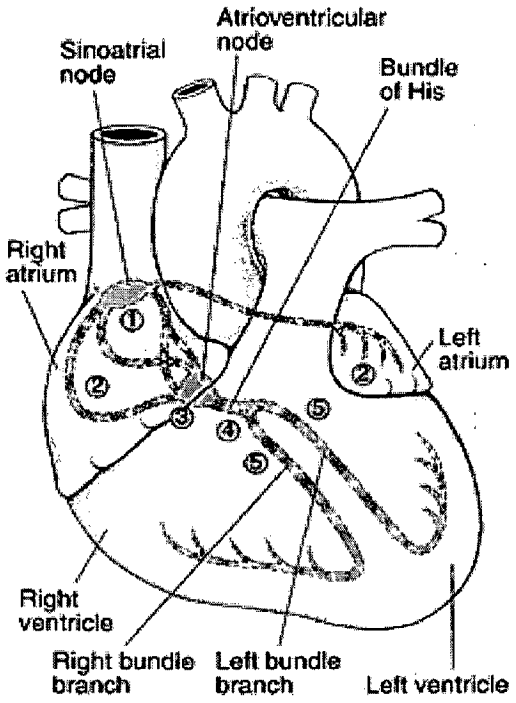



Figure2.8 Electrical Pathway of the Heart [38]

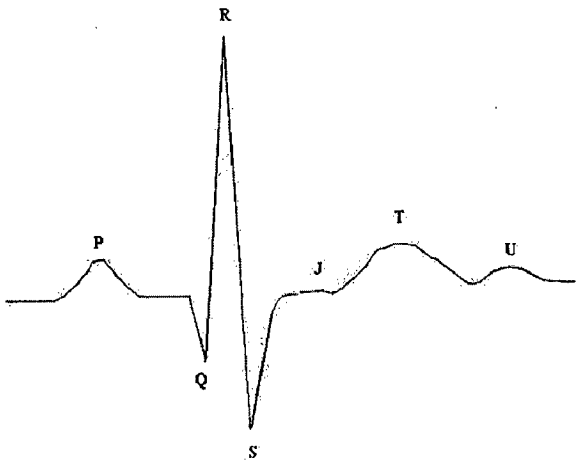



Figure 2.9 The Electrocardiogram

The sinoatrial node (1) initiates an electrical impulse that flows through the right and left atria (2), making them contract. When the electrical impulse reaches the atrioventricular node (3), it is delayed slightly. The impulse then travels down the bundle of His (4), which divides into the right bundle branch for the right ventricle (5) and the left bundle branch for the left ventricle (5). The impulse then spreads through the ventricles, making them contract.[38]

In synchronized excitation of these specialized cells, the electrical charged tend to migrate to the body fluids. Such charge migration constitutes an electric current and hence setup potential differences between various points of the body including the outer surfaces. Such potential difference can be conveniently picked up by placing electrodes at any points on the surface of the body and measured with the help of sensitive instruments, and results in the waveform electrocardiogram. See figure 2.2.

## 2.3 Mechanical Aspects of ECG signal

The contraction and relaxation of a cardiac chamber are respectively known as systole and diastole. The movements of the cardiac chambers are related in a cycle manner during each heart beat that constitutes a cardiac cycle [1] .

The phase during which both atria and ventricles are in diastole and are released simultaneously is called the joint diastole. During this phase, blood continues to flow in to the atria through the Superior and Inferior Vena cava. At the end of this phase, the next heart beat starts with the contraction of atria. As the atria contracts, they force most of their blood to the ventricles which are still in diastole. The atria act as a pump to collect and force the venous blood to the ventricles. During atrial diastole, blood cannot pass back from the atria into the great veins, because the roots of veins are compressed by the atrium contraction to block their opening. [2].

At the end of the atrial systole, the atrium relapses into diastole and starts releasing during the atrial systole. Venous blood again passes from great veins to atria to fill them up. Simultaneously, the ventricles start contracting (ventricular systole). The pressure rises immediately in the ventricles to exceed that in the atria and the AV valves are shut sharply to prevent back flow of blood from the ventricles to the atria.

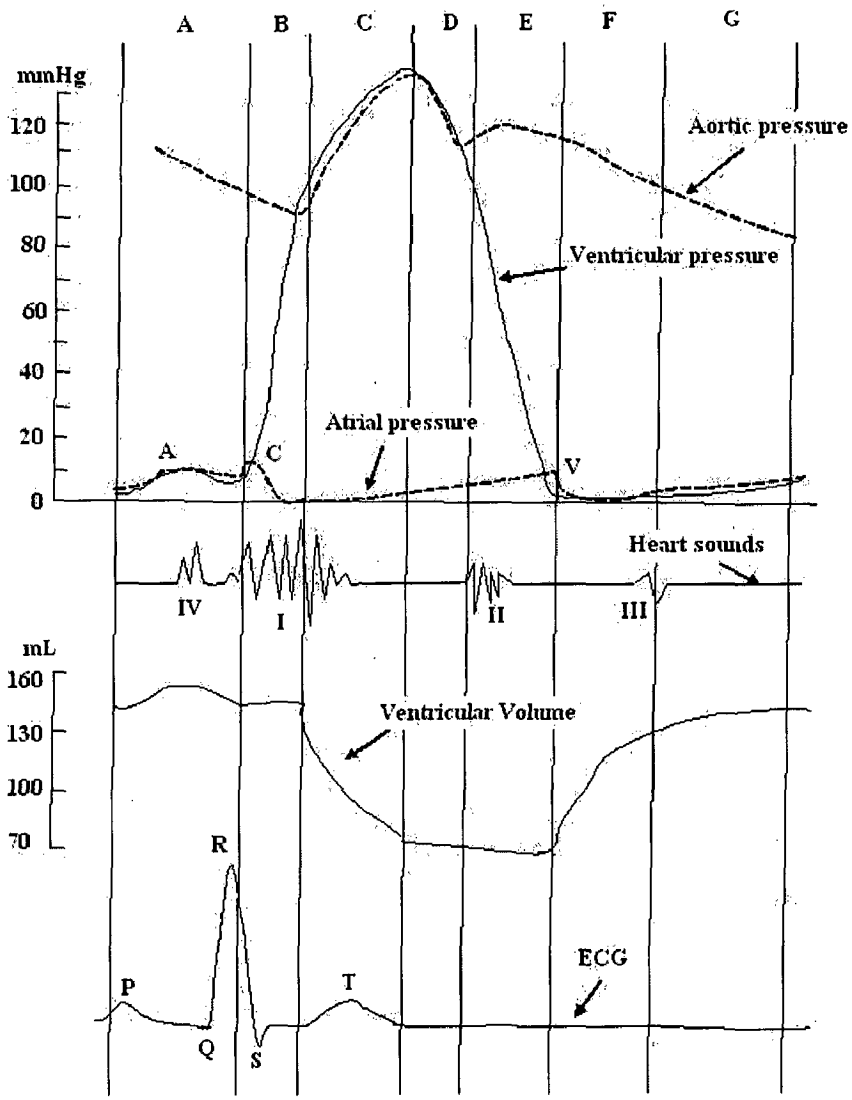



**Figure 2.3 Mechanical and electrical events during a single cardiac cycle. The seven phases are denoted by letters as follows: A) atrial systole, (B) isovolumetric ventricular contraction, (C) rapid ventricular ejection, (D) reduced ventricular ejection, (E) isovol isovolumetric ventricular relaxation, (F) rapid ventricular filling, and (G) reduced ventricular filling.[39]**

This sharp closure of AV valves at the beginning of the ventricular systole produces a sound "Lub" in the heart. This is the first heart sound during a heart beat which can be heard by placing a stethoscope on the chest wall above the heart. Because the ventricular pressure still continues to be lower than the pressure in the great arteries and semi lunar valve consequently remains closed, the ventricles now contract as closed chambers. But as the ventricular systole progresses the pressure in them increases rapidly and soon exceeds the pressure in the great arteries. Semi-lunar valve now open and blood begins to flow into the great arteries.

At the end of ventricular systole, the ventricles go into diastole and start relaxing (ventricular diastole) immediately the semi-lunar valve closes sharply to prevent back flow of blood from the great arteries to the ventricles. The closure of the semi-lunar valves at the beginning of the ventricular diastole produces a sound "Dub" in the heart. This is the second heart sound [1] [2]. The physical activities of the heart with their corresponding ECG component are shown in fig 2.3.

## 2.4 Basic ECG components

Before performing any kind of signal processing on ECG signals it is necessary to study its basic components. If these components are altered in an irrational way it may result in wrong interpretation of the ECG data. The normal and abnormal conditions of each component are discussed in detail in this section.

An ECG is composed of a series of waves and lines usually ordered into some repeatable pattern. The waves and lines are displayed on either a two dimensional screen or on ECG paper. The height of the tracing represents millivolts while the width of the ECG addresses an interval of time (see Figure 2.4).

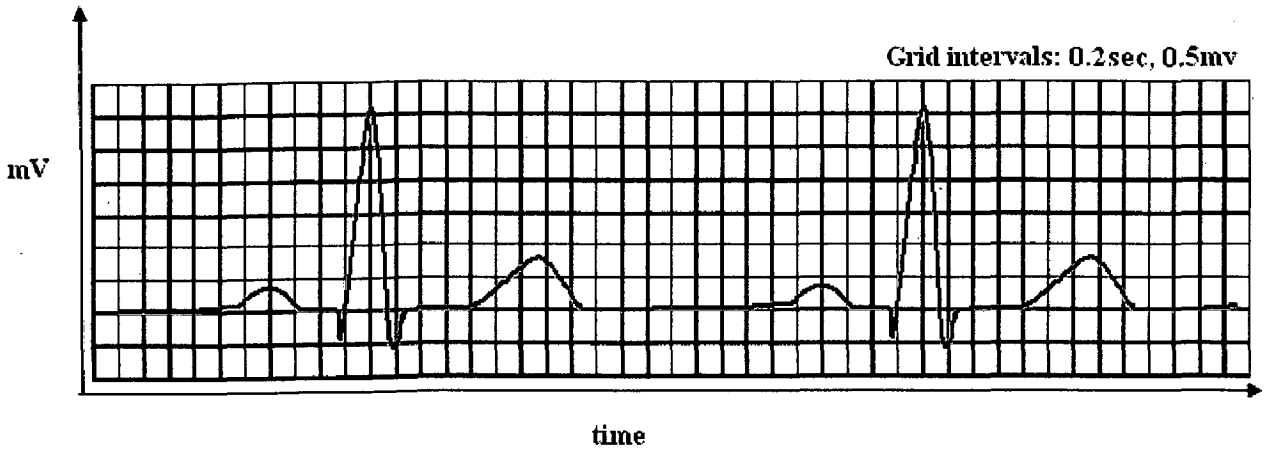



**Figure2.4 The scales of the ECG**

This section on ECG components addresses each of the waves, intervals and segments of an ECG in the order that they would appear. While waves are fairly self-explanatory, intervals measure time from the start of one wave to the start of another wave (an interval includes at least one wave) and segments measure time between waves (waves are not included in a segment). Table 2.1 outlines the parameters that are expected of normal and abnormal ECG waves, segments and intervals.

### The P Wave, PR Segment and PR Interval

The P wave represents the depolarization of the right and left atria. The P wave begins with the first deviation from baseline and finishes when the wave meets the baseline once again. While the P wave is an electrical representation and not mechanical, a P wave strongly suggests that the atria have followed through with a contraction. [3]

Table 2.7 Normal and Abnormal Parameters of ECG Components [3]

| ECG Components | Normal Parameters | Abnormal Parameters | Causes of abnormal Parameters |
|---|---|---|---|
| P Wave | Upright in most leads including lead II.<br>Duration: < 0.11 seconds<br>Amplitude: 0.5-2.5 sec. | Inverted<br><br>Notched or tall | Junctional Rhythm<br><br>Atrial rhythm, atrial hypertrophy |
| PR Interval | Duration: 0.12 - 0.20 sec. | Duration: shorter or longer than normal | Junctional rhythm, Wolff-Parkinson-White syndrome |
| Q Wave | Duration: <0.04 seconds<br>Amplitude: <25% the amplitude of the R wave | Duration: 0.04 sec. or longer<br>Amplitude: at least 25% the amplitude of the R wave | Myocardial infarction |
| QRS Complex | Upright, inverted or biphasic waveform<br>Duration: < 0.11 seconds<br>Amplitude: 1 mm or more | Duration: 0.11 second or more | Bundle branch block, ventricular ectopic i.e. PVC |
| QT Interval | Duration: less than 1/2 the width of the R-R interval | Duration: at least 1/2 the R-R interval | Long QT syndrome, cardiac drugs, hypothermia, subarachnoid hemorrhage<br><br>Short QT associated with hypercalcemia |
| ST Segment | In line with PR or TP segment (baseline)<br>Duration: shortens with increased heart rate | Deviation of 0.5 mm or more from baseline | Cardiac ischemia or infarction, early repolarization, ventricular hypertrophy, digoxin dip, pericarditis, subarachnoid hemorrhage |
| T Wave | Upright, asymmetrical and bluntly rounded in most leads<br>Duration: 0.10-0.25 sec.<br>Amplitude: less than 5 mm | Peaked, inverted, biphasic, notched, flat or wide waveforms | Cardiac ischemia or infarction, subarachnoid hemorrhage, left-sided tension pneumothorax, left bundle branch block, hyperkalemia, hypokalemia |
| U Wave | Upright<br>Amplitude: < 2 mm | Peaked or Inverted<br>Amplitude: > 2 mm | Hypokalemia, cardiomyopathy, ventricular hypertrophy, diabetes, digoxin, quinidine |

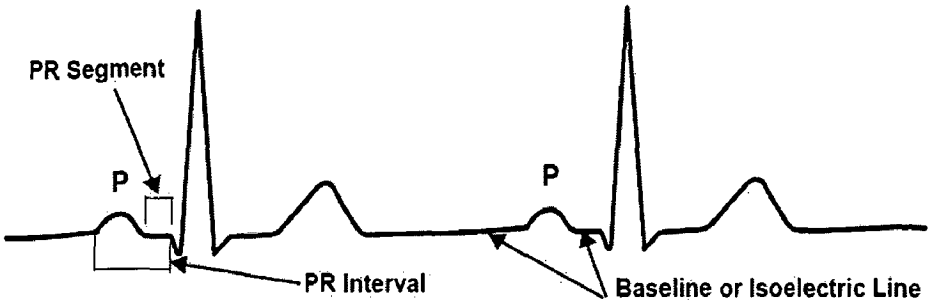



Figure 2.5 The P Wave, PR Segment and PR Interval [3]

The PR segment is the line between the end of the P wave and the beginning of the QRS complex. The PR segment signifies the time taken to conduct through the slow AV junction. This delay allows for atrial kick. The PR segment also serves as a benchmark for the isoelectric line. [3]

The PR interval is measured from the start of the P wave to the start of the QRS complex. While it might appear obvious that this is indeed a PQ interval, a Q wave is not always present on an ECG tracing. For consistency, the term is PR interval has been adopted whether a Q wave exists or not.

The PR interval covers the time taken for the impulse to travel from the SA node through the atria and the AV junction through to the Purkinje network. Most of the PR interval is taken by the slow conducting AV junction. Changes to the PR interval often points to the AV junction. A normal PR interval is 0.12-0.20 seconds, which is the equivalent to 3-5 small squares (3-5mm) on ECG paper.

If an ECG shows P wave, QRS complex - P wave, QRS complex - P wave, QRS complex - atrial depolarization, ventricular depolarization until the cows come home, a rather important relationship between the atria and the ventricle is revealed. If the P wave is consistently followed

by a QRS complex across a consistent PR interval, this is strong evidence that the originating impulse is supraventricular. A consistent PR interval is often sufficient to declare that this is a supraventricular rhythm. [3]

**The QRS Complex**

ECG interpretation relies heavily on the QRS complex. The QRS complex represents the depolarization of the ventricles [3]. The repolarization of the atria is also buried in the QRS complex.

The normal depolarization of the ventricles is illustrated in Figure 4.13 on the next page. Three distinct waveforms are often present in a normal QRS complex. These waveforms follow the pathways of ventricular depolarization. Depolarization of the ventricular septum proceeds first from left to right away from the positive electrode in lead II [3]. This early depolarization causes a small downward deflection called a Q wave.

A Q wave is the first negative deflection of the QRS complex that is not preceded by an R wave [3]. A normal Q wave is narrow and small in amplitude. Note that a wide and/or deep Q wave may signify a previous myocardial infarction (MI). More on the signs of cardiac ischemia and infarction is addressed in the next section.

Following the depolarization of the interventricular septum, ventricular depolarization then progresses from the endocardium through to the epicardium across both ventricles producing an R wave and an S wave. An R wave is the first positive deflection of the QRS complex [3]. An S wave is the first wave after the R wave that dips below the baseline (isoelectric line). The end of the S wave occurs where the S wave begins to flatten out. This is called the J point.

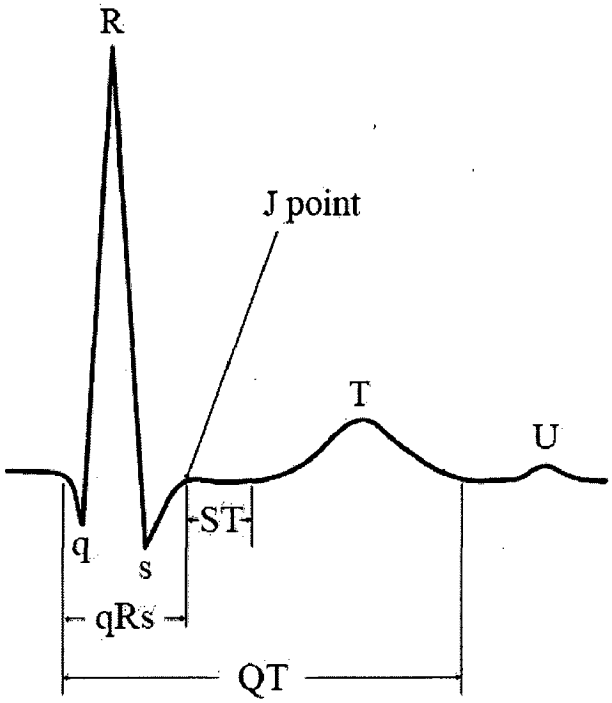



**Figure 2.6 The QRS Complex, ST Segment and the T Wave [3]**

Abnormal ventricular depolarization produces a QRS complex that often has additional waveforms. For example, a second positive deflection of a QRS complex after an R wave is labelled R' (R prime). Similarly, a second S wave that dips below the baseline after the R wave is labelled S' (S prime).

A narrow QRS complex occurs quickly over a period of less than 0.11 seconds (less than 3 mm in width) [3]. A narrow QRS occurs with normal ventricular depolarization that originates above the ventricles.

While the direction of the QRS complex is generally not important with basic ECG interpretation, the width of the QRS complex is key. The width of the QRS complex often indicates the location of the originating electrical impulse. This is a rather important point since the first and foremost word of an ECG interpretation is the location of impulse initiation.

For example, rhythms that come from the SA node are sinus rhythms, from the AV junction are junctional rhythms, and that originate from the ventricle are ventricular rhythms. Simple. If the QRS is narrow - taking very little time to occur - the cardiac rhythm originates from a supraventricular site. Quickly determining whether the QRS is narrow or wide is a vital step in rapid ECG interpretation.

**The Q Wave and The QT Interval**

As mentioned in the previous section, a normal Q wave represents a depolarization of the ventricular septum, which usually travels from left to right, towards the right ventricle. When present, a Q wave is the first downward deflection of the QRS complex. While ST segment deviation is a sign of present events, a prominent Q wave points to an MI that has already occurred, recently to some time ago. A prominent Q wave is like a tattoo - once you have one, it's pretty much yours for good [3].

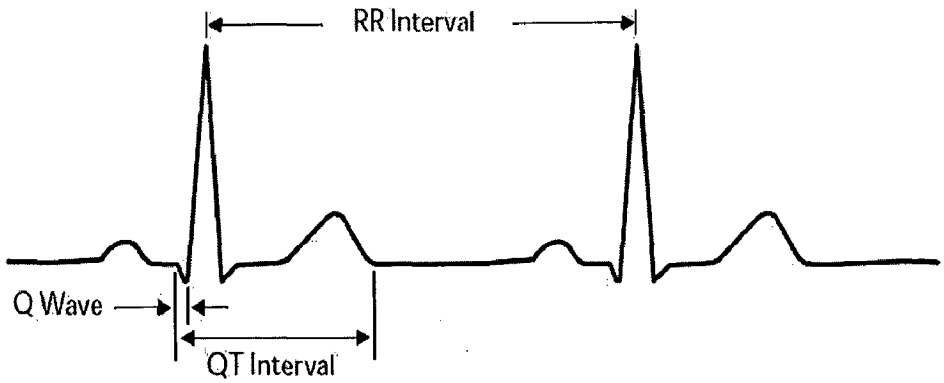



Figure 2.7 The Normal Q Wave and QT Interval [3]

A normal Q wave is usually no deeper than 2 mm and less than 1 small square in width (<0.04 seconds). An abnormal Q wave tends to get the most attention [3]. A Q wave that is wider

than 1 small square or at least 1/4 the height of the R wave is a significant marker of a myocardial infarction.

The QT interval represents a complete ventricular cycle of depolarization and repolarization [3]. The QT interval is measured from the beginning of the QRS complex to the end of the T wave. A QT interval should be less than 1/2 the R-R interval.

The concern around a longer QT interval centers around the possibility of the next QRS coming at the tail end of the T wave, called an R-on-T phenomenon. This phenomenon can potentially cause dangerous dysrhythmias such as torsades de pointes. Causes of prolonged QT intervals include long QT syndrome, antiarrythmics such as quinidine and procainamide, tricyclic antidepressants, and hypokalemia.

**The ST Segment**

Between the QRS complex and the T wave, lies the ST segment. The ST segment usually follows the isoelectric line. The ST segment represents early repolarization of the ventricles [3]. Explained in detail in Chapter 6, early repolarization includes a plateau phase where the cardiac cell membrane potential does not change.

During early repolarization, the positive ion potassium exits the cardiac cell while the positive ion calcium enters the cardiac cell, effectively negating any change in cell membrane potential. Because the cell membrane does not change its electrical potential, ECG leads do not record any electrical activity [3]. As a result, the ST segment usually lies along the ECG baseline.

Determining where the ST segment begins is determined by the J point. The J point, the juncture of the QRS and the ST segment, defines the starting point of the ST segment. The J point marks where the QRS complex changes direction, forming a notch or bump in the ECG tracing [3]. The ST segment is evaluated for any deviation from the ECG baseline 0.04 seconds (1 mm) after the J point.

While ST deviations may be a normal occurrence for a subset of the population, most often ST deviation is a sign of either myocardial ischemia, myocardial infarction and/or cardiac disease. It makes sense, then, to report any finding of ST deviation from baseline in the ECG interpretation i.e. sinus rhythm with ST depression.

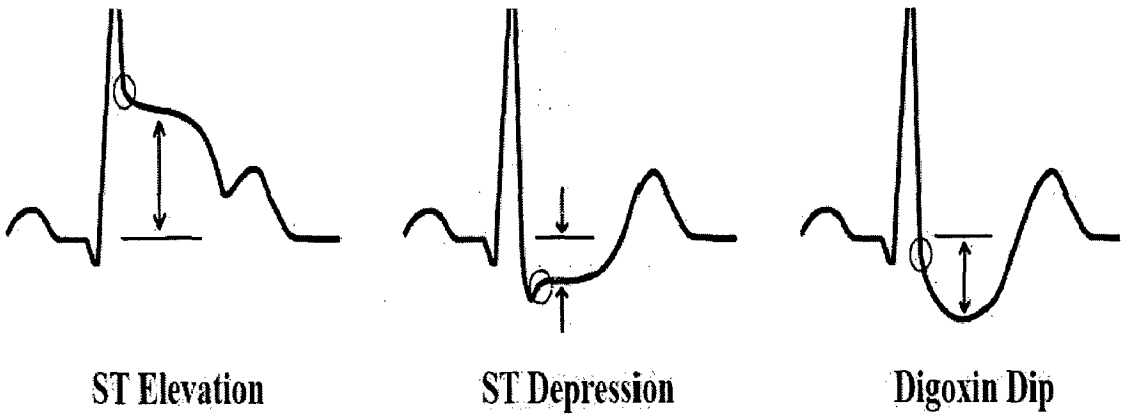



Figure 2.8 ST Segment Deviations [3]

ST depression of 1 mm or more in 2 contiguous leads (neighboring leads) is suggestive of myocardial ischemia, injury or infarction. ST elevation of 1 mm or more in 2 contiguous leads is highly suggestive of a myocardial injury or infarction [3]. Note that ST changes (elevation or depression) are highly suggestive of current events - the acute coronary events are happening now.

The shape of the ST segment, if depressed, bears mention. The depressed ST segment often presents horizontal, sloping downwards or sloping upwards. Although all morphologies can indicate myocardial ischemia, the horizontal and downward sloping depressed ST segments are the more likely morphologies that point to ischemic events.

Several conditions not linked to cardiac ischemia can produce ST changes. The bottom line: most ST changes indicate cardiac ischemia, requiring urgent treatment BUT every ECG interpretation is more robust when integrated with a patient's clinical status and history.

**The T Wave**

Expect a T wave to follow every QRS complex. The T wave is a graphic representation of the repolarization of the ventricle [3]. The T wave is typically about 0.10 to 0.25 seconds wide with an amplitude less than 5 mm. While ventricular depolarization occurs rapidly producing a tall QRS complex, ventricular repolarization is spread over a longer interval, resulting in a shorter and broader T wave.

The T wave is normally slightly asymmetrical and is usually larger than the P wave. The T wave is normally upright in lead II. Note that as heart rates increase, the P wave (atria) and the T wave (ventricles) begin to share the same space on an ECG. The larger T wave often covers the P wave. Note that the T wave is rarely notched. A notched T wave may also contain a P wave trying to show itself [3].

An inverted T wave can point to cardiac ischemia among other causes. Ischemia to the epicardium prolongs ventricular repolarization to this area. This extended repolarization of the epicardium removes the delay between the repolarization of the endocardium and the repolarization of the epicardium, with repolarization now following the sequence of depolarization. An inverted T wave results. [3]

Abnormally shaped T waves can signify acute episodes of cardiac ischemia, electrolyte imbalances, and the influence of cardiac medications. For example, peaked T waves can occur early during periods of myocardial ischemia and infarction. Later, cardiac ischemia may cause the T wave to invert [3]. Electrolyte imbalances can also affect the T wave. Hyperkalemia is often associated with peaked T waves. Hypokalemia can flatten the T wave. Quinidine can widen the T wave while digitalis can flatten the T wave.

Abnormally shaped T waves can also occur following injury to the lungs or the brain. While the physiology is not well understood, T wave inversion can occur with a left-sided tension pneumothorax [3]. Peaked or inverted T waves have also been reported with brain injury, specifically subarachnoid hemorrhage.

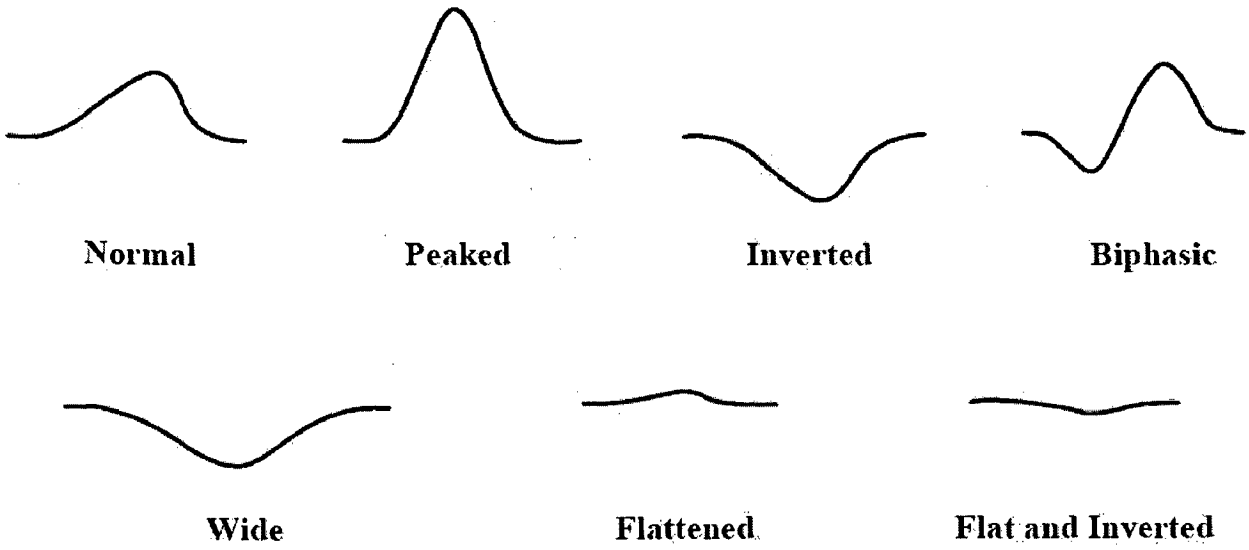



Figure 2.9 Normal and Abnormal T Waves [3]

## The U Wave

Occasionally, another wave -the U wave - is recorded immediately following the T wave and before the P wave. The U wave remains rather mysterious but is thought to represent a final

stage of repolarization of unique ventricular cells in the midmyocardium [3]. The U wave will most often orient in the same direction as the T wave with an amplitude less than 2 mm.

An abnormal U wave is inverted or tall with an amplitude of 2 mm or more. An abnormally tall U wave is associated with conditions such as hypokalemia, diabetes, ventricular hypertrophy, and cardiomyopathy. Cardiac medications such as digoxin and quinidine can also cause a tall U wave.

The U wave the series of waves, intervals and segments that form the ECG. Knowing what to expect from each the these components prepares you to quickly recognize deviations from the norm.

## 2.5 ECG leads and lead selection

As the heart undergoes depolarization and repolarization, electrical currents spread throughout the body because the body acts as a volume conductor. The electrical currents generated by the heart are commonly measured by an array of electrodes placed on the body surface. By convention, electrodes are placed on each arm and leg, and six electrodes are placed at defined locations on the chest. These electrode leads are connected to a device that measures potential differences between selected electrodes to produce the characteristic ECG tracings.

Some of the ECG leads are bipolar leads (e.g., standard limb leads) that utilize a single positive and a single negative electrode between which electrical potentials are measured. Unipolar leads (augmented leads and chest leads) have a single positive recording electrode and utilize a combination of the other electrodes to serve as a composite negative electrode. Normally, when an ECG is recorded, all leads are recorded simultaneously, giving rise to what is called a 12-lead ECG. [4]

**Standard Limb Leads (Bipolar)**

Leads I, II and III are the so-called limb leads because at one time, the subjects of electrocardiography had to literally place their arms and legs in buckets of salt water in order to obtain signals for Einthoven's string galvanometer [35]. They form the basis of what is known as Einthoven's triangle. Eventually, electrodes were invented that could be placed directly on the patient's skin. Even though the buckets of salt water are no longer necessary, the electrodes are still placed on the patient's arms and legs to approximate the signals obtained with the buckets of salt water. They remain the first three leads of the modern 12 lead ECG.

- Lead I is a dipole with the negative (white) electrode on the right arm and the positive (black) electrode on the left arm.
- Lead II is a dipole with the negative (white) electrode on the right arm and the positive (red) electrode on the left leg.
- Lead III is a dipole with the negative (black) electrode on the left arm and the positive (red) electrode on the left leg.

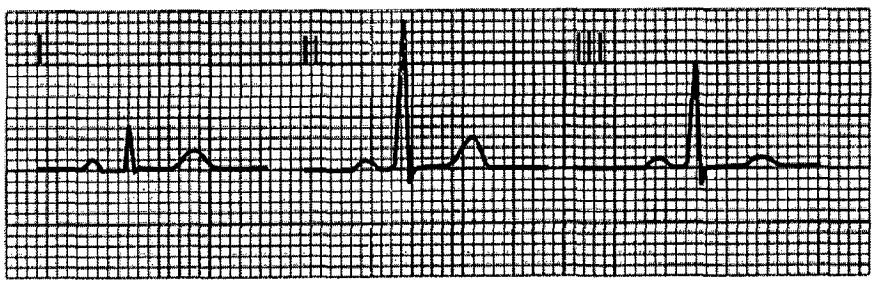



**Figure 2.10 Bipolar Limb Leads [4]**

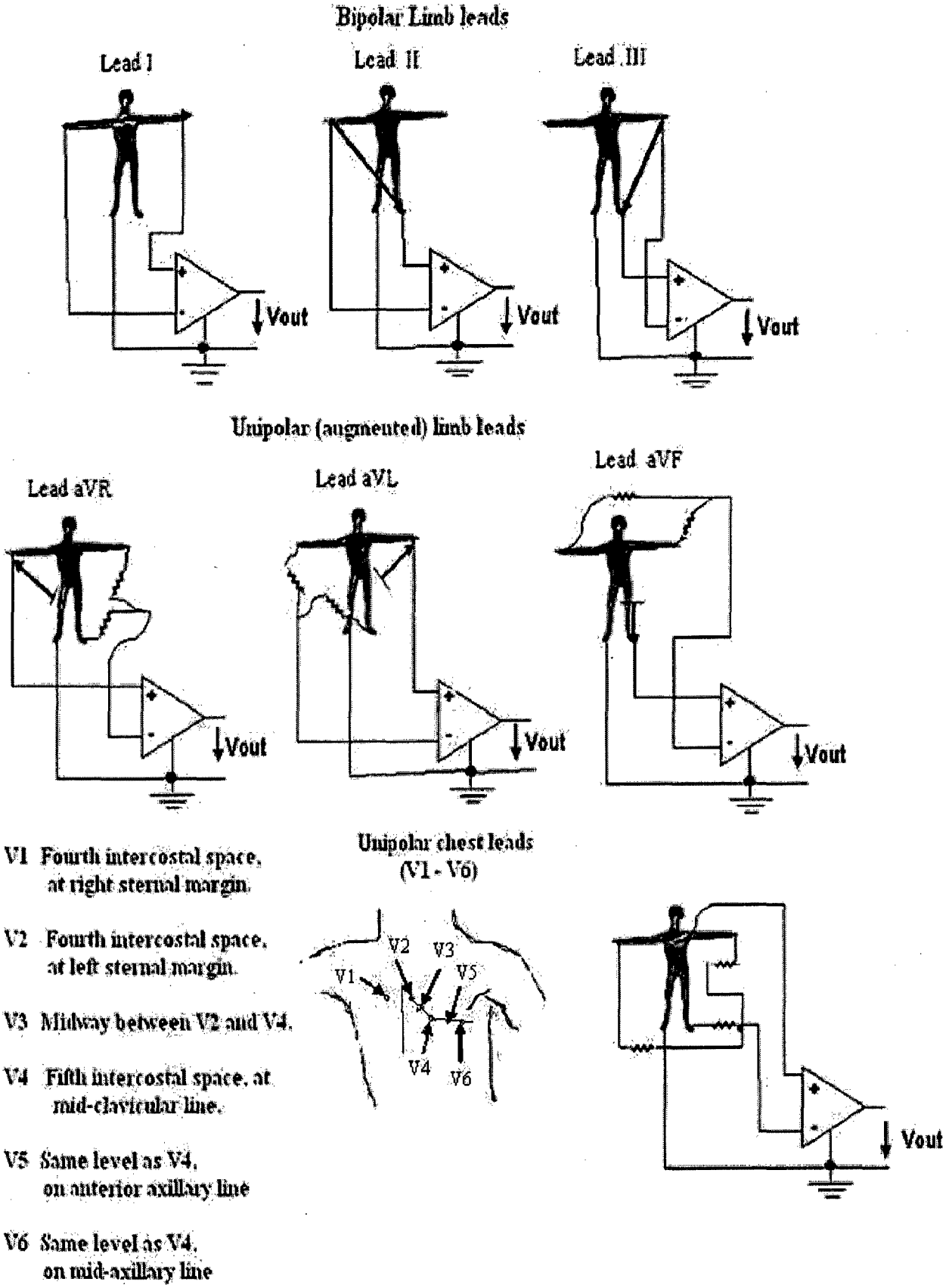



Figure 2.11 ECG lead configurations [2]

**Augmented limb leads**

Leads aVR, aVL, and aVF are augmented limb leads. They are derived from the same three electrodes as leads I, II, and III. However, they view the heart from different angles (or vectors) because the negative electrode for these leads is a modification of Wilson's central terminal, which is derived by adding leads I, II, and III together and plugging them into the negative terminal of the EKG machine. This zeroes out the negative electrode and allows the positive electrode to become the "exploring electrode" or a unipolar lead. This is possible because Einthoven's Law states that I + (-II) + III = 0. The equation can also be written I + III = II. It is written this way (instead of I + II + III = 0) because Einthoven reversed the polarity of lead II in Einthoven's triangle, possibly because he liked to view upright QRS complexes. Wilson's central terminal paved the way for the development of the augmented limb leads aVR, aVL, aVF and the precordial leads V1, V2, V3, V4, V5, and V6. [35]

- Lead aVR or "augmented vector right" has the positive electrode (white) on the right arm. The negative electrode is a combination of the left arm (black) electrode and the left leg (red) electrode, which "augments" the signal strength of the positive electrode on the right arm.
- Lead aVL or "augmented vector left" has the positive (black) electrode on the left arm. The negative electrode is a combination of the right arm (white) electrode and the left leg (red) electrode, which "augments" the signal strength of the positive electrode on the left arm.
- Lead aVF or "augmented vector foot" has the positive (red) electrode on the left leg. The negative electrode is a combination of the right arm (white) electrode and the left

arm (black) electrode, which "augments" the signal of the positive electrode on the left leg.

The augmented limb leads aVR, aVL, and aVF are amplified in this way because the signal is too small to be useful when the negative electrode is Wilson's central terminal. Together with leads I, II, and III, augmented limb leads aVR, aVL, and aVF form the basis of the hexaxial reference system, which is used to calculate the heart's electrical axis in the frontal plane. [35]

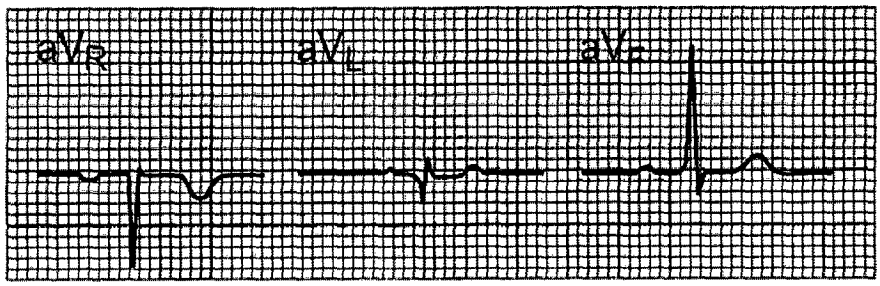



Figure 2.12 Augumented Limb Leads [4]

**Precordial chest leads**

The precordial leads V1, V2, V3, V4, V5, and V6 are placed directly on the chest. Because of their close proximity to the heart, they do not require augmentation. Wilson's central terminal is used for the negative electrode, and these leads are considered to be unipolar. The precordial leads view the heart's electrical activity in the so-called horizontal plane. The heart's electrical axis in the horizontal plane is referred to as the Z axis. [35]

Leads V1, V2, and V3 are referred to as the right precordial leads and V4, V5, and V6 are referred to as the left precordial leads.

The QRT complex should be negative in lead V1 and positive in lead V6. The QRT complex should show a gradual transition from negative to positive between leads V2 and V4.

The equiphasic lead is referred to as the transition lead. When the transition occurs earlier than lead V3, it is referred to as an early transition. When it occurs later than lead V3, it is referred to as a late transition. There should also be a gradual increase in the amplitude of the R wave between leads V1 and V4. This is known as R wave progression. Poor R wave progression is a nonspecific finding. It can be caused by conduction abnormalities, myocardial infarction, cardiomyopathy, and other pathological conditions.[35]

- Lead V1 is placed in the fourth intercostal space to the right of the sternum.
- Lead V2 is placed in the fourth intercostal space to the left of the sternum.
- Lead V3 is placed directly between leads V2 and V4.
- Lead V4 is placed in the fifth intercostal space in the midclavicular line (even if the apex beat is displaced).
- Lead V5 is placed horizontally with V4 in the anterior axillary line
- Lead V6 is placed horizontally with V4 and V5 in the midaxillary line.

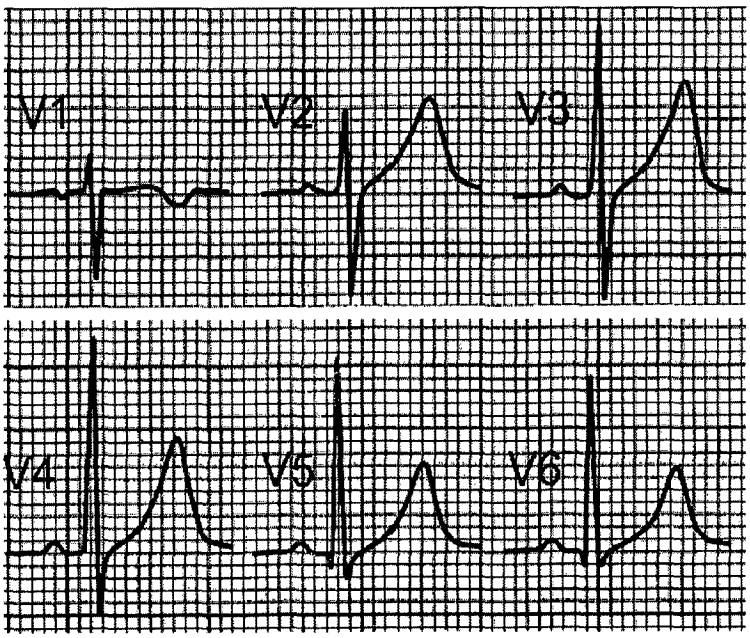



**Figure2.12 Pericordial Chest Leads [4]**

**The selected Lead configuration for the proposed work**

It is possible to use any of the above lead configurations to study the application of artificial neural network to ECG data compression. In this work the most commonly used bipolar limb lead II (LII) configuration is selected. However any data can be compressed by making use of the system developed.

## 2.6 Conclusion

Electrocardiograms (ECGs) are biopotentials generated by the muscles of the heart. The electrical currents resulted by these potentials are commonly measured by an array of electrodes placed on the body surface.

A comprehensive knowledge about each component of an ECG signal gives a chance to investigate the effect of the compression system, to be developed, on the original data. Defects seen on the reconstructed data can only be identified as "real" abnormalities or compression system effects if we have the knowledge of Each ECG component's characteristics.

ECG signals are quasi-periodic, i.e. they have a sequence of cycles which are very much similar with only little changes. This is an important feature because if we are going to use artificial neural networks, we can store the little differences in the hidden layers and represent the similarities by a single weight. The next chapter gives a detail explanation about these networks; which brought a paradigm shift in the computation world.

# CHAPTER THREE

# ARTIFICIAL NEURAL NETWORK – AN OVERVIEW

## 3.1 Introduction

In the previous chapter we have discussed about the basic characteristic of the electrocardiogram (ECG) and its prominent feature, its quasi-periodicity, which led to the usage of artificial neural networks to compress this signals. In this chapter we will discuss in about these computational tools.

An Artificial Neural Network (ANN) is a massively parallel distributed processor made up of simple processing units, which has a natural propensity for storing experiential knowledge and making it available for use [9]. It is an information processing paradigm that is inspired by the way biological nervous systems, such as the brain, process information. It is composed of a large number of highly interconnected processing elements (neurons) working in unison to solve specific problems.

An ANN is configured for a specific application, such as pattern recognition, data classification or data compression, through a learning process. Learning in biological systems involves adjustments to the synaptic connections that exist between the neurons [10]. This is true of ANNs as well.

## 3.2 Historical background [5]

Neural network simulations appear to be a recent development. However, this field was established before the advent of computers, and has survived at least one major setback and several eras.

The history of neural networks can be divided into several periods:

1. **First Attempts:** There were some initial simulations using formal logic. McCulloch and Pitts (1943) developed models of neural networks based on their understanding of neurology. These models made several assumptions about how neurons worked. Two groups (Farley and Clark, 1954; Rochester, Holland, Haibit and Duda, 1956). The first group (IBM reserchers) maintained closed contact with neuroscientists at McGill University. So whenever their models did not work, they consulted the neuroscientists. This interaction established a multidisciplinary trend which continues to the present day.

2. **Promising & Emerging Technology:** Not only was neuroscience influential in the development of neural networks, but psychologists and engineers also contributed to the progress of neural network simulations. Rosenblatt (1958) stirred considerable interest and activity in the field when he designed and developed the Perceptron. The Perceptron had three layers with the middle layer known as the association layer. This system could learn to connect or associate a given input to a random output unit. Another system was the ADALINE (ADAptive LInear Element) which was developed in 1960 by Widrow and Hoff (of Stanford University). The ADALINE was an analogue electronic device made from simple components. The method used for learning was different to that of the Perceptron, it employed the Least-Mean-Squares (LMS) learning rule.

3. **Period of Frustration & Disrepute:** In 1969 Minsky and Papert wrote a book in which they generalised the limitations of single layer Perceptrons to multilayered systems. In the book they said: "...our intuitive judgment that the extension (to multilayer systems) is sterile". The significant result of their book was to eliminate funding for research with neural network simulations. The conclusions supported the disenhantment of reserchers in the field. As a result, considerable prejudice against this field was activated.

4. **Innovation:** Although public interest and available funding were minimal, several researchers continued working to develop neuromorphically based computaional methods for problems such as pattern recognition. During this period several paradigms were generated which modern work continues to enhance.Grossberg's (Steve Grossberg and Gail Carpenter in 1988) influence founded a school of thought which explores resonating algorithms. They developed the ART (Adaptive Resonance Theory) networks based on biologically plausible models. Anderson and Kohonen developed associative techniques independent of each other. Klopf (A. Henry Klopf) in 1972, developed a basis for learning in artificial neurons based on a biological principle for neuronal learning called heterostasis.

   Werbos (Paul Werbos 1974) developed and used the back-propagation learning method, however several years passed before this approach was popularized. Back-propagation nets are probably the most well known and widely applied of the neural networks today. In essence, the back-propagation net. is a Perceptron with multiple layers, a different thershold function in the artificial neuron, and a more robust and capable learning rule. Amari (A. Shun-Ichi 1967) was involved with theoretical developments: he published a paper which established a mathematical theory for a learning basis (error-correction

method) dealing with adaptive patern classification. While Fukushima (F. Kunihiko) developed a step wise trained multilayered neural network for interpretation of handwritten characters. The original network was published in 1975 and was called the Cognitron.

5. **Re-Emergence:** Progress during the late 1970s and early 1980s was important to the re-emergence on interest in the neural network field. Several factors influenced this movement. For example, comprehensive books and conferences provided a forum for people in diverse fields with specialized technical languages, and the response to conferences and publications was quite positive. The news media picked up on the increased activity and tutorials helped disseminate the technology. Academic programs appeared and courses were inroduced at most major Universities (in US and Europe). Attention is now focused on funding levels throughout Europe, Japan and the US and as this funding becomes available, several new commercial with applications in industry and finacial institutions are emerging.

6. **Today:** Significant progress has been made in the field of neural networks-enough to attract a great deal of attention and fund further research. Advancement beyond current commercial applications appears to be possible, and research is advancing the field on many fronts. Neurally based chips are emerging and applications to complex problems developing. Clearly, today is a period of transition for neural network technology.

## 3.3 Neuron model, transfer functions and network architecture

### 3.3.1 Neuron Model: Single-Input Neuron

A single-input neuron is shown in Figure 3.1. The scalar input $p$ is multiplied by the scalar weight $w$ to form $wp$, one of the terms that are sent to the summer. The other input,$1$ , is multiplied by a bias $b$ and then passed to the summer. The summer output $n$, often referred to as the net input, goes into a transfer function $f$, which produces the scalar neuron output $a$ .

If we relate this simple model back to the biological neuron, the weight corresponds to the strength of a synapse, the cell body is represented by the summation and the transfer function and the neuron output $a$ represents the signal on the axon. [6]

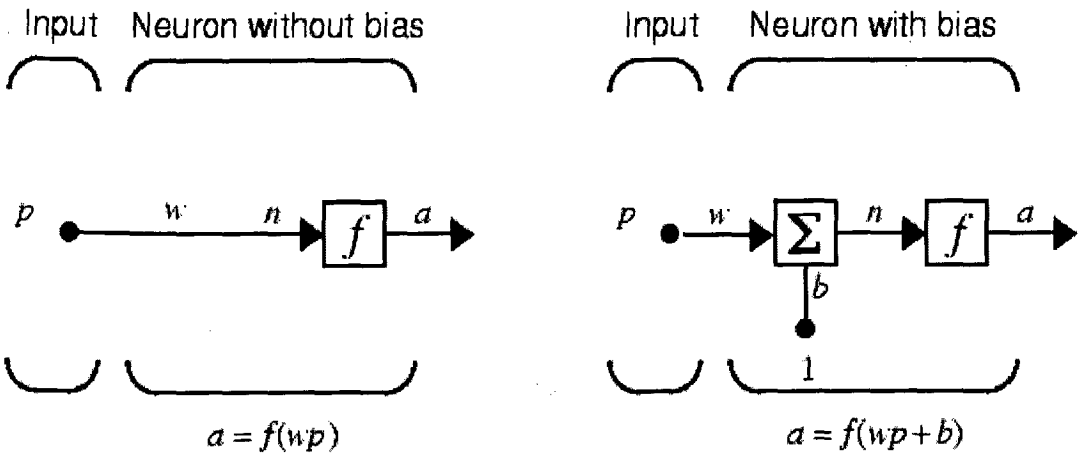



Figure 3.1 Single input neuron [7]

The actual output depends on the particular transfer function that is chosen.

### 3.3.2 Transfer Functions

Many transfer functions are used in artificial Neural Network. Three of the most commonly used functions are shown below.

***Hard-Limit Transfer Function***

The hard-limit transfer function shown above limits the output of the neuron to either 0, if the net input argument *n* is less than 0, or 1, if *n* is greater than or equal to 0 [7].

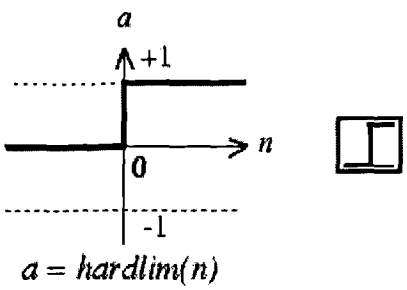



Figure 3.2 hard-limit transfer function

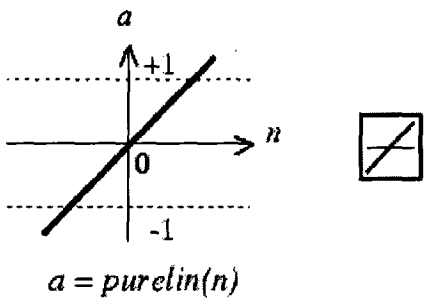



Figure 3.3 linear transfer function

*Linear Transfer Function*

Figure 3.3 illustrates the linear transfer function. Neurons of this type are used as linear approximators in Linear Filters.

*Sigmoid transfer function*

The sigmoid transfer function shown below takes the input, which can have any value between plus and minus infinity, and squashes the output into the range -1 to 1. This transfer function is commonly used in backpropagation networks, in part because it is differentiable.

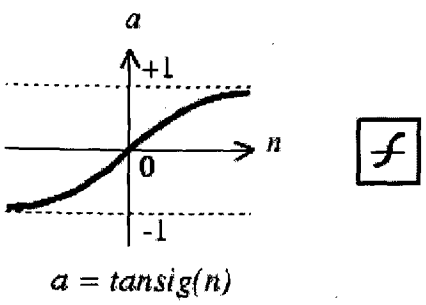
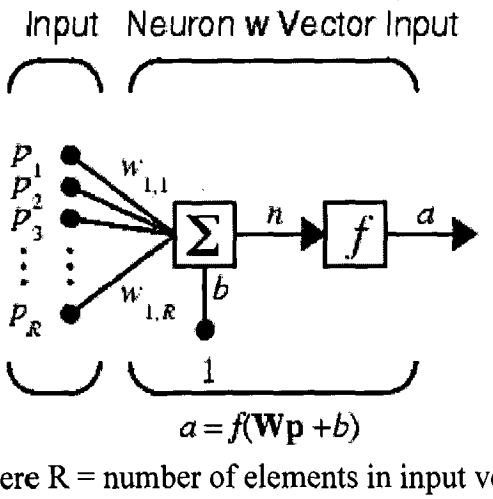



Figure 3.10 sigmoid transfer function




where R = number of elements in input vector

Figure 3.11 Neuron with Vector Input [7]

*Neuron with Vector Input*

A neuron with a single R-element input vector is shown below. Here the individual element inputs $p_1, p_2, \cdots, p_R$ are multiplied by weights $w_{1,1}, w_{1,2}, \cdots, w_{1,R}$ and the weighted values are fed to the summing junction. Their sum is simply $wp$, the dot product of the (single row) matrix $W$ and the vector $p$. [6]

The neuron has a bias $b$, which is summed with the weighted inputs to form the net input $n$. This sum, $n$, is the argument of the transfer function $f$.

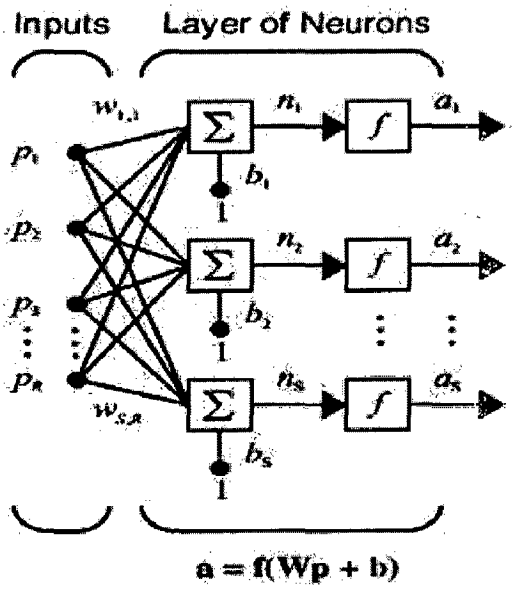
$$n = w_{1,1}p_1 + w_{1,2}p_2 + \cdots + w_{1,R}p_R$$

### 3.3.3 Network Architectures

Two or more of the neurons shown earlier can be combined in a layer, and a particular network could contain one or more such layers. First consider a single layer of neurons.

*A Layer of Neurons*

A one-layer network with $R$ input elements and $S$ neurons is shown below.




where R = number of elements in input vector

S = number of neurons in a layer

Figure 3.12 A layer of neurons [7].

In this network, each element of the input vector $p$ is connected to each neuron input through the weight matrix $W$. The $i^{th}$ neuron has a summer that gathers its weighted inputs and

bias to form its own scalar output $n(i)$. The various $n(i)$ taken together form an $S$-element net input vector $n$. Finally, the neuron layer outputs form a column vector $a$.

### *Multiple Layers of Neurons*

A network can have several layers. Each layer has a weight matrix $W$, a bias vector $b$, and an output vector $a$. To distinguish between the weight matrices, output vectors, etc., for each of these layers in the figures, the number of the layer is appended as a superscript to the variable of interest. You can see the use of this layer notation in the three-layer network shown below, and in the equations at the bottom of the figure.

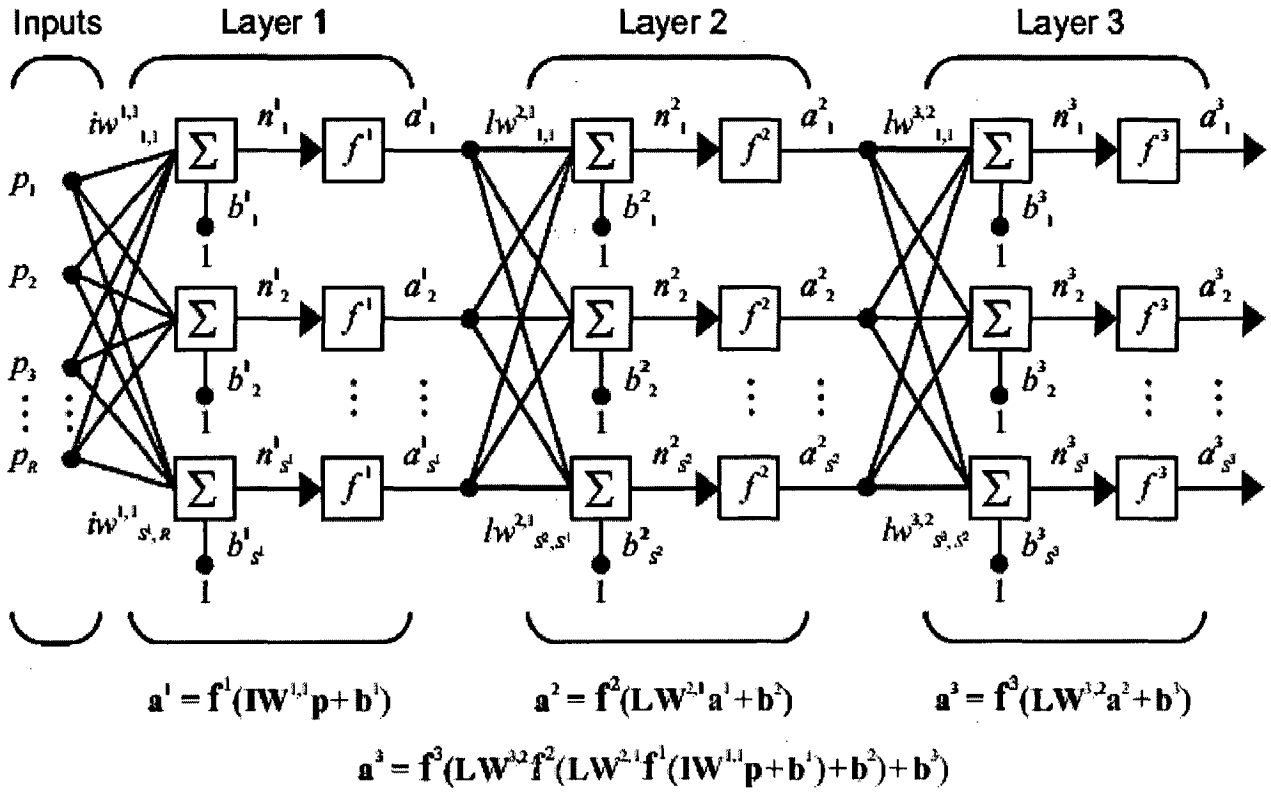



**Figure 3.13 Multiple Layers of Neurons [7].**

The network shown above has $R^1$ inputs, $S^1$ neurons in the first layer, $S^2$ neurons in the second layer, etc. It is common for different layers to have different numbers of neurons. A constant input 1 is fed to the bias for each neuron.

## 3.4 Perceptrons, feedforward networks and backpropagation algorithm

***Perceptrons***

An arrangement of one input layer of McCulloch-Pitts neurons feeding forward to one output layer of McCulloch-Pitts neurons is known as a Perceptron. It is similar to the layer of neurons discussed above.

***Feed-forward networks***

Feed-forward ANNs allow signals to travel one way only; from input to output. There is no feedback (loops) i.e. the output of any layer does not affect that same layer. Feed-forward ANNs tend to be straight forward networks that associate inputs with outputs. They are extensively used in pattern recognition. This type of organization is also referred to as bottom-up or top-down [5].

***Multi-Layer Perceptron (MLP)***

Multi-layer perceptrons are feed-forward nets with one or more layers of nodes between the input and output nodes. These additional layers contain hidden units or nodes that are not directly connected to both the input and output nodes. [8]

Multi-layer perceptrons have been applied successfully to solve some difficult and diverse problems in training them in a supervised manner with a highly popular algorithm, known as the error back-propagation algorithm. The algorithm is based on the error-correction learning rule.

The multi-layer perceptron neural network is much more powerful than the single layer network. The MLP has three distinctive characteristics [9]:

1) Each neuron includes a nonlinear activation function, which makes the nonlinearity of the input-output relation possible.

2) The network contains one or more layers of hidden neurons, which enable the network to learn complex tasks by extracting more meaningful features from the input patterns.
3) The network exhibits a high degree of connectivity, determined by the synapses of the network.

The multi-layer perceptron neural network model consists of a network of processing elements or nodes arranged in layers. Typically, it requires three or more layers of processing nodes: an input layer which accepts the input variables used in the classification procedure, one or more hidden layers, and an output layer with one node for one class.

The principle of MLP is that when data from an input pattern is presented at the input layer the network nodes perform calculations in the successive layers until an output value is computed at each of the output nodes [9].

***Backpropagation algorithm***

The back propagation training algorithm is an iterative gradient algorithm designed to minimize the mean square error between the actual output of a multilayer feed-forward perceptron and the desired output. It requires continuous differentiable non-linearities. The following assumes a sigmoid logistic non-linearity is used where the function $f(\alpha)$ is [8]

$$f(\alpha) = \frac{1}{1 + e^{-(\alpha - \theta)}} \tag{3.1}$$

The back-propagation algorithm consists of five steps [8]:

Step1. Initialize weights and offsets

Set all weights and node offsets to small random values.

Step2. Present input and desired outputs

Present a continuous valued input vector $x_0, x_1, \ldots, x_{N-1}$ and specify the desired outputs $d_0, d_1, \ldots, d_{M-1}$.

Step3. Calculate actual outputs

Use the sigmoid nonlinearity from above and hidden activation and weights values to calculate outputs $y_0, y_1, \ldots, y_{M-1}$.

Step4. Adapt weights

Use a recursive algorithm starting at the output nodes and working back to the first hidden layer. Adjust weights by

$$w_{ij}(t+1) = w_{ij}(t) + \eta \delta_j x_i' \tag{3.2}$$

In this equation $w_{ij}(t)$ is the weight from hidden mode $i$ or from an input to node $j$ at time $t$, $x_i'$ is either the output of node $i$ or is an input, $\eta$ is a gain term, and $\delta_j$ is an error term for node $j$. if node $j$ is an output node, then

$$\delta_j = y_j(1-y_j)(d_j - y_j), \tag{3.3}$$

Where $d_j$ is the desired output of node $j$ and $y_j$ is the actual output.

If node j is an internal hidden node, then

$$\delta_j = x_j'(1-x_j')\sum_k \delta_k w_{jk}, \tag{3.4}$$

Where $k$ is overall nodes in the layers above node $j$. Internal node thresholds are adapted in a similar manner by assuming they are connection weights on links from auxiliary constant-valued inputs.

Step5. Repeat by going to step2

## 3.5 The learning process

All learning methods used for adaptive neural networks can be classified into two major categories:

- **Supervised learning** which incorporates an external teacher, so that each output unit is told what its desired response to input signals ought to be. During the learning process global information may be required. Paradigms of supervised learning include error-correction learning, reinforcement learning and stochastic learning [5].

  An important issue concerning supervised learning is the problem of error convergence, i.e. the minimization of error between the desired and computed unit values. The aim is to determine a set of weights which minimizes the error. One well-known method, which is common to many learning paradigms, is the least mean square (LMS) convergence.

- **Unsupervised learning** uses no external teacher and is based upon only local information. It is also referred to as self-organization, in the sense that it self-organizes data presented to the network and detects their emergent collective properties. Paradigms of unsupervised are Hebbian learning and competitive learning. [5]

## 3.6 Conclusion

An Artificial Neural Network (ANN) is a massively parallel distributed processor made up of highly interconnected simple processing elements called neurons. The way this processor works is inspired by the way biological nervous systems, such as the brain, process information.

If we relate ANN neuron to the biological neuron, the weight corresponds to the strength of a synapse, the cell body is represented by the summation and the transfer function and the neuron output represents the signal on the axon.

Two or more of the neurons can be combined in a layer, and a particular network could contain one or more such layers. A special kind of layered neurons is the multi-layer perceptron (MLP). MLPs are feed-forward nets with one or more layers of nodes between the input and output nodes. They have been applied successfully to solve some difficult and diverse problems if trained in a supervised manner with a highly popular algorithm, known as the back-propagation algorithm.

The back propagation training algorithm is an iterative gradient algorithm designed to minimize the mean square error between the actual output of a multilayer feed-forward perceptron and the desired output.

If we use a three layer perceptron and train it in a supervised manner using a robust backpropagation algorithm, it can be applied to ECG data compression. The idea is to represent the small differences between ECG beats using a hidden layer and the similarities using a single weight. If the number of hidden neurons are less than the number of the input neurons then data are compressed.

Before proceeding to present the application of this technique to compress ECG data, we will discuss, in the next chapter, the existing ECG data compression schemes and view their merits and demerits. This will be crucial in evaluation and comparison of the proposed technique with others.

# CHAPTER FOUR

# EXISTING ECG DATA COMPRESSION TECHNIQUES

## 4.1 Introduction

In chapter two, we have discussed about the heart of cardiac monitoring systems, the ECG signal. We have selected artificial neural networks as our computational tools in order to compress these signals. Thus we have given an overview of these computational tools in chapter 3. But before presenting my proposed technique of ECG data compression, I would like to discuss in this chapter about the existing data compression techniques which are applied to the ECG signal. This will help us to make a fair comparison between the proposed technique and the existing ones.

Data compression techniques have been utilized in a broad spectrum of communication areas such as speech, image, and telemetry transmission [11]. Data compression methods have been mainly classified into three major categories [12]: a) direct data compression, b) transformation methods, and c) parameter extraction techniques.

Data compression by the transformation or the direct data compression methods contains transformed or actual data from the original signal. Whereby, the original data are reconstructed by an inverse process. The direct data compressors base their detection of redundancies on direct analysis of the actual signal samples. In contrast, transformation compression methods mainly utilize spectral and energy distribution analysis for detecting redundancies.

On the other hand, the parameter extraction method is an irreversible process with which a particular characteristic or parameter of the signal is extracted. The extracted parameters (e.g., measurement of the probability distribution) are subsequently utilized for classification based on a priori knowledge of the signal features.

Existing data compression techniques for ECG signals lie in two of the three categories described: the direct data and the transformation methods. Direct data compression techniques for ECG signals have shown a more efficient performance than the transformation techniques in regard particularly to processing speed and generally to compression ratio [13]. Most of the transformation techniques have been developed specifically for data compression of multi-orthogonal ECG leads.

Each compression scheme is presented in accordance to the following five issues [11]:

a) A brief description of the structure and the methodology behind each ECG compression scheme is presented along with any reported unique advantages and disadvantages.
b) The issue of processing time requirement for each scheme has been excluded. In light of the current technology, all ECG compression techniques can be implemented in real-time environments due to the relatively slow varying nature of ECG signals.
c) The sampling rate and precision of the ECG signals originally employed in evaluating each compression scheme are presented along with the reported compression ratio.
d) Since most of the databases utilized in evaluating ECG compression schemes are nonstandard, database comparison has been excluded. We believe such information does not provide additional clarity and at times may be misleading. However, every effort has been made to include comments on how well each compression scheme has performed.

The intent is to give the reader a feeling for the relative value of each compression technique.

e) Finally, the fidelity measure of the reconstructed signal compared to the original ECG has been primarily based on visual inspection. Besides the visual comparison, many compression schemes have employed the percent root-mean-square difference (PRD). The PRD calculation is as follows:

$$PRD = \sqrt{\frac{\sum_{i=1}^{n}[x_{org}(i) - x_{rec}(i)]^2}{\sum_{i=1}^{n} x_{org}{}^2(i)}} * 100 \qquad (4.1)$$

where $x_{org}$ and $x_{rec}$ are samples of the original and reconstructed data sequences.

## 4.2 Classical direct data compression methods

Most of the direct data compression techniques rely on utilizing prediction or interpolation algorithms. These techniques attempt to reduce redundancy in a data sequence by examining a successive number of neighboring samples. A prediction algorithm utilizes *a priori* knowledge of some previous samples, while an interpolation algorithm employs *a priori* knowledge of both previous and future samples.

Direct data compression methods are classified into three categories: tolerance comparison Compression, differential pulse code modulation (DPCM), and entropy coding methods

### 4.2.1 Tolerance-Comparison Data Compression Techniques

Most of the tolerance-comparison data compression techniques employ polynomial predictors and interpolators. The basic idea behind polynomial prediction/ interpolation compressors is to eliminate samples, from a data sequence, that can be implied by examining preceding and succeeding samples. The implementation of such compression algorithms is

usually executed by setting a preset error threshold centered on an actual sample point. Whenever the difference between that sample and a succeeding future sample exceeds the preset error threshold, the data between the two samples is approximated by a line whereby only the line parameters (e.g., length and amplitude) are saved. Description of tolerance-comparison compression techniques based on polynomial predictors/interpolators has been consolidated in [14]

a) Polynomial Predictors: Polynomial predictors are based on a finite difference technique which constraints an nth-order polynomial to K + 1 data points. Predicted data are obtained by extrapolating the polynomial one sample point at a time. The polynomial predictor [15] [16] is

$$\hat{y}_n = y_{n-1} + +\Delta^2 y_{n-1} + \cdots + \Delta^k y_{n-1} \tag{4.2}$$

Where $\hat{y}_n$ = predicted sample point at time *t*,

$y_{n-1}$ = sample value at one sample period prior to *t*,

$\Delta y_{n-1} = y_{n-1} - y_{n-2}$

$\Delta^k y_{n-1} = \Delta^{k-1} y_{n-1} - \Delta^{k-1} y_{n-2}$

The value of *k* represents the order of the polynomial prediction algorithm.

*Zero-Order Predictor (ZOP):* The ZOP is a polynomial predictor with $k = 0$. In this case,

$$\hat{y}_n = y_{n-1} \tag{4.3}$$

where the predicted value is merely the previous data point. Several implementations of this algorithm are exploited by employing different aperture (peak error) techniques [12], [16],

[17]. The most efficient ZOP technique uses a floating aperture (sometimes called the step method) wherein a tolerance band $\pm\epsilon$ is centered on the last saved data point as shown in Figure 4.1.Succeeding sample points that lie in the tolerance band ($\pm\epsilon$) , centered around the last saved sample point, are not retained. The tolerance band actually ''floats" with the non-redundant (saved) data points.

*First-Order Predictor (FOP):* The FOP is an implementation of equation (4.2) with $k = 1$ [12], [16], [17]. This yields a first-order polynomial of the form,

$$\hat{y}_n = 2y_{n-1} - y_{n-2} \tag{4.4}$$

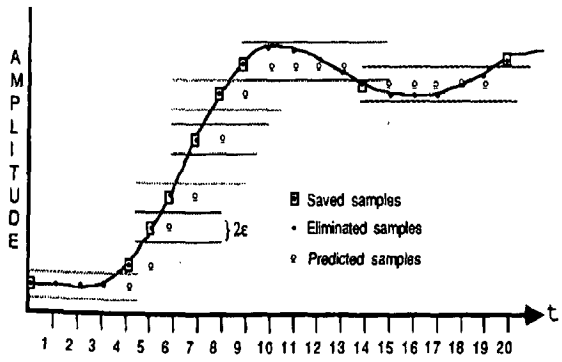



Figure 4.1 Illustration of the ZOP floating aperture[11]

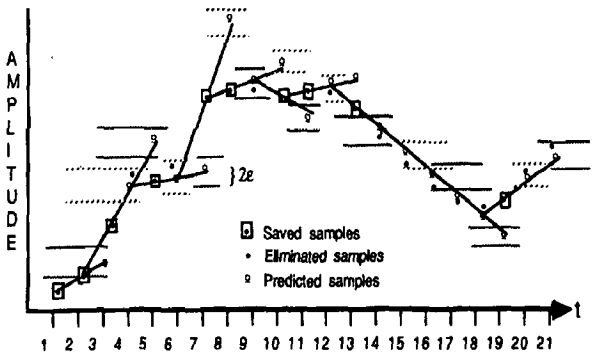



Figure 4.2 Illustration of the FOP floating aperture [11]

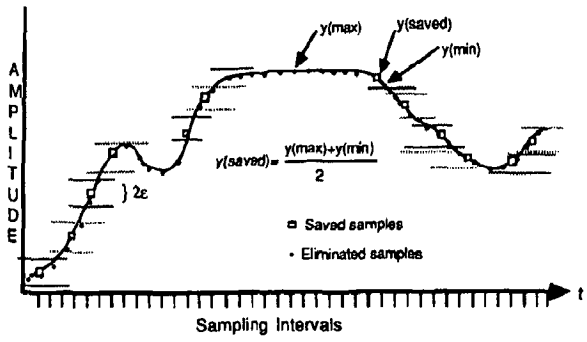



Figure 4.3 Illustration of the Zero-order interpolator. [11]

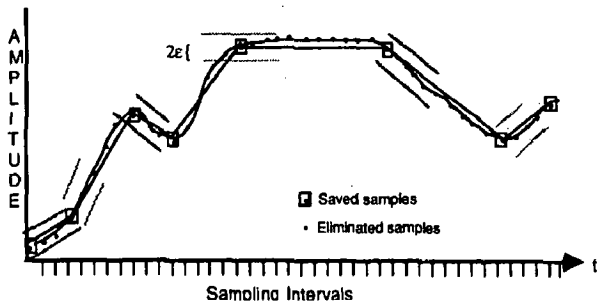



Figure 4.4 Principal operation of the FOI-2DF [11]

The predicted value is a point on the straight line drawn between the last two data points ($y_{n-1}$ and$y_{n-2}$). The FOP algorithm with a floating aperture (Figure 4.2) is initiated by retaining the first two data points and drawing a straight line between these two points. An aperture of width $\pm\epsilon$ is centered on the obtained line. If the actual sample point ($y_n$) is within $\pm\epsilon$ of the predicted value, then that sample point is not saved. Otherwise, ($y_n$) is saved and a new prediction line is drawn through ($y_n$) and the previous predicted point. The signal reconstruction requires the non-redundant sample values along with the corresponding time.

b) Polynomial Interpolators: Unlike the case of prediction, polynomial interpolators utilize both past and future data points to decide whether or not the actual sample point is redundant. In other words, all samples between the last retained sample and the present sample point affect the interpolation. Low-order polynomial interpolators have been found to be very efficient in ECG data compression [18].

*Zero-Order Interpolator (ZOI):* The principal operation of the zero-order interpolator is illustrated in Fig. 3. The ZOI is similar to the ZOP in the sense that a horizontal (zero-order) line is employed to determine the largest set of consecutive data points within a preset error

threshold. The main difference lies in selecting the sample point that represents the redundant set. The interpolator retained sample is determined at the end of the redundant set, in contrast to the first sample in the case of the predictor. Moreover, the saved sample for the interpolator algorithm is computed as the average between the minimum and the maximum sample values in the set. [11]

*First-Order Interpolator (FOI):* The first-order interpolator (linear method) assumes that data will continue in the same direction (slope) once it has started. Instead of drawing a horizontal line as is the case in the zero-order method, a line is drawn to establish a slope. The firstorder interpolator with two degrees of freedom (FOI-2DF) has been found to be the most efficient compression scheme among other first-order interpolators [12], [19]. The FOI-2DF draws a straight line between the present sample and the last saved sample so that intermediate data points are within a specified tolerance of the interpolated value. The encoded message contains information about the length of the line and its starting and ending points. The ending point of a line, in this interpolation scheme, is used as the starting point of the next line segment. This results in a reduced code word length with decreased flexibility (i.e., two degrees of freedom). In other words, only one data point (the ending point) needs to be retained for each line after the very first saved line.

### 4.2.2 Data Compression by Differential Pulse Code Modulation

The basic idea behind the differential pulse code modulation (DPCM) is that when data samples are estimated, the error (residual) between the actual sample and the estimated sample value($e_n = y_n - \hat{y}_n$) is quantized and transmitted or stored [20]. Consequently, waveform redundancy reduction by DPCM coders is basically achieved by representing the actual correlated signal, in terms of an uncorrelated signal, namely, the estimation error signal.

### 4.2.3 Entropy Coding

The theoretical basis of entropy coding can be traced back to Shannon's theorem of communication theory [21]. Data compression by entropy coding is obtained by means of assigning variable-length code words to a given quantized data sequence according to their frequency of occurrence. This compression method attempts to remove signal redundancy that arises whenever the quantized signal levels do not occur with equal probability.

## 4.3 Direct ECG Data Compression Schemes

This section presents the direct data compression schemes developed specifically for ECG data compression.

### 4.3.1 The AZTEC Technique

The amplitude zone – time epoch coding (AZTEC) algorithm originally developed by Cox et al. [22] for preprocessing real-time ECG's for rhythm analysis. It has become a popular data reduction algorithm for ECG monitors and databases with an achieved compression ratio of 10: 1 (500 Hz sampled ECG with 12 b resolution) [23] [24]. However, the reconstructed signal demonstrates significant discontinuities and distortion. In particular, most of the signals distortion occurs in the reconstruction of the P and T waves due to their slow varying slopes.

The AZTEC algorithm converts raw ECG sample points into plateaus and slopes. The AZTEC plateaus (horizontal lines) are produced by utilizing the zero-order interpolation (ZOI) discussed above .b). The stored values for each plateau are the amplitude value of the line and its length (the number of samples with which the line can be interpolated within aperture E). The production of an AZTEC slope starts when the number of samples needed to form a plateau is less than three. The slope is saved whenever a plateau of three samples or more can be formed. The stored values for the slope are the duration (number of samples of the slope) and the final

elevation (amplitude of last sample point). Signal reconstruction is achieved by expanding the AZTEC plateaus and slopes into a discrete sequence of data points.

### 4.3.2 The Turning Point Technique

The turning point (TP) data reduction algorithm [24] was developed for the purpose of reducing the sampling frequency of an ECG signal from 200 to 100 Hz without diminishing the elevation of large amplitude QRS's.

The algorithm processes three data points at a time; a reference point ($X_0$) and two consecutive data points ($X_1$ and $X_2$). Either XI or X2 is to be retained. This depends on which point preserves the slope of the original three points. The TP algorithm produces a fixed compression ratio of 2:1 whereby the reconstructed signal resembles the original signal with some distortion. A disadvantage of the TP method is that the saved points do not represent equally spaced time intervals.

### 4.3.3 The CORTES Scheme

The coordinate reduction time encoding system (CORTES) algorithm [26] is a hybrid of the AZTEC and TP algorithms. CORTES applies the TP algorithm to the high frequency regions (QRS complexes), whereas it applies the AZTEC algorithm to the isoelectric regions of the ECG signal. The AZTEC and TP algorithms are applied in parallel to the incoming sampled ECG data. Whenever an AZTEC line is produced, a decision based on the length of the line is used to determine whether the AZTEC data or the TP data is to be saved. If the line is longer than an empirically determined threshold, the AZTEC line is saved; otherwise the TP data are saved. Only AZTEC plateaus (lines) are generated; no slopes are produced. The CORTES signal reconstruction is achieved by expanding the AZTEC plateaus into discrete data points and interpolating between each pair of the TP data.

Performance evaluation of the AZTEC, TP, and CORTES algorithms were reported in [26] (ECG's sampled at 200 Hz with 12 b resolution) with compression ratios of 5 : 1, 2 : 1, and 4.8 : 1 respectively, and PRD's of 28, 5, and 7, respectively.

### 4.3.4 Fan and SAPA Techniques

Fan and scan-along polygonal approximation (SAPA) algorithms, developed for ECG data compression, are based on the first-order interpolation with two degrees of freedom (FOI-2DF) technique.

a) The Fan Algorithm*:* In essence, the Fan is a method of implementing the FOI-2DF without requiring the storage of all the actual data points between the last transmitted point and the present point during program execution. Moreover, it draws the longest possible line between the starting point and the ending point so that all intermediate samples are within the specified error tolerance [11]. An illustration of the Fan method is shown in Fig.4.6. The Fan algorithm starts by accepting the first data point as a nonredundant (permanent) point $(t_o)$ and functions as the origin. Two slopes $(U_1, L_1)$ are drawn between the originating point and the next sample plus a specified threshold $(\pm\epsilon)$. One upper slope $(U_1)$ passes through a point greater than the second sample point value by a tolerance $(\epsilon)$, while the other lower slope $(L_1)$ passes through a point less than the second sample point value by an $\epsilon$. If the third sample point $(t_2)$ falls within the area bounded by the two slopes, then new slopes $(U_2, L_2)$ are calculated between the originating point and an $\epsilon$ greater and an $\epsilon$ lower than the third sample point. These new slopes $(U_2, L_2)$ are compared to the previously stored slopes $(U_1, L_1)$ and the most converging (restrictive) slopes are retained $(U_2, L_2)$. [11]

The process is repeated whereby future sample values are compared with the values of the most convergent slopes. Whenever a sample value falls outside the area bounded by the converging slopes, the sample immediately preceding this sample point is saved as the next permanent sample. This permanent sample point also becomes the new originating point and the algorithm repeats.

b) SAPA-2 Algorithm: The theoretical bases of this algorithm are that the deviation between the straight lines (approximated signal) and the original signal is never more than the preset error tolerance *(ϵ)*. The only difference between the Fan and SAPA-2

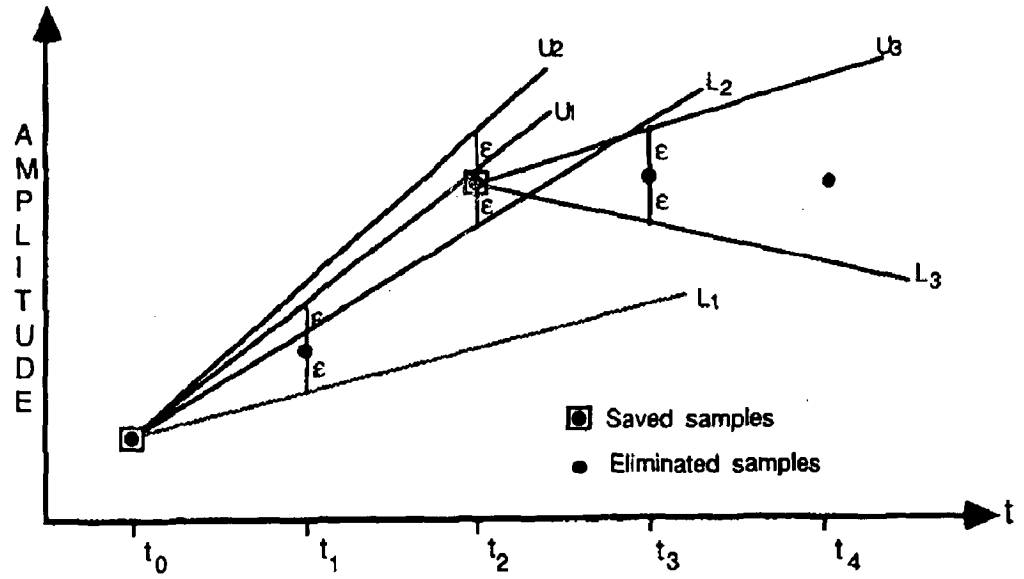



**Figure 4.6 Illustration of the Fan method. Upper and lower slopes ( U and L ) are drawn within threshold ($\epsilon$) around sample points taken at $t_1$, $t_2$, etc [11]**

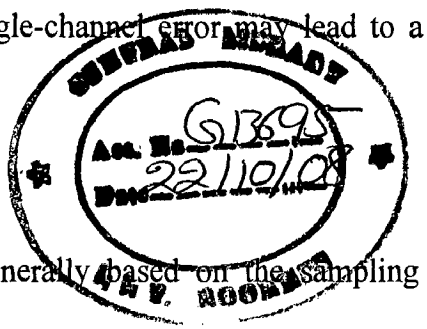
algorithms is that, in addition to the two slopes calculated in the Fan algorithm, SAPA- 2 calculates a third slope between the originating sample point and the actual future sample point (called center slope). Whenever the center slope value does not fall within the two converging slopes boundary, the immediately preceding sample point is considered as a permanent sample. [11]

### 4.3.5 ECG Data Compression by DPCM

The simplest DPCM system for data compression is a system that employs the predictor $\hat{y}_n = y_{n-1}$. Hence, the first-difference signal (amplitude between successive samples $e_n = y_n - \hat{y}_n$ is substituted for the actual signal itself. ECG data compression based on such a system has been referred to as "delta coding." [11]

### 4.3.6 Entropy Coding of ECG's

The output of an ECG DPCM encoder is, however, mapped into variable length codewords instead of fixed length ones. A disadvantage of variable length encoding is the possibility of serious decoding errors that may occur due to transmission errors [11]. If the codewords are not delimited by special means, a single-channel error may lead to a long sequence of erroneous receiver outputs.




### 4.3.7 Peak-Picking Compression of ECG's

The peak-picking compression techniques are generally based on the sampling of a continuous signal at peaks (maxima and minima) and other significant points of the signal. The basic operation of such techniques involves the extraction of signal parameters that convey "most" of the signal information. These parameters include the amplitude and location of the maxima and the minima points, slope changes, zero-crossing intervals, and points of inflection in the signal. These parameters are substituted in place of the original signal. Upon reconstruction, the signal is restored by polynomial fitting techniques such as straight lines or parabolic functions. [11]

### 4.3.8 ECG Cycle-to-Cycle Compression

Basically, the rationale of the cycle-to-cycle compression method is to substitute a periodic signal by one cycle period and a count of the total number of cycles that occur in the

signal. Yet this approach is only applicable to periodic signals with the constraint that all the signal cycles are exactly the same, which is not the case in ECG waveforms. However, the ECG is a quasi-periodic signal which does not change appreciably in morphology except as a result of a change in the heart function. The cycle-to-cycle ECG compression technique may potentially result in a high compression ratio when applied to Holter ECG's. This is best justified by noting that in the case of Holter ECG's [27] only certain short-period segments of the 24 h recording show abnormality relative to the large number of normal sinus ECG's.

**Table 4.1 Summary of some ECG data compression schemes[11]**

| Method | Compression ratio | signal frequency (Hz) | signal precision (bits) | PRD (%) | comments |
|---|---|---|---|---|---|
| AZTEC | 10.0 | 500 | 12 | 28.0 | Poor P and T Fidelity |
| TP | 2.0 | 200 | 12 | 5.3 | Sensitive to signal frequency(SF) |
| CORTES | 4.8 | 200 | 12 | 7.0 | Sensitive (SF) Poor P Fidelity |
| Fan/SAPA | 3.0 | 250 | -- | 4.0 | High Fidelity |
| Entropy Coding of $2^{nd}$-difference ECGs | 2.8 | 250 | 10 | -- | susceptible to transmission errors |
| peak-picking (spline) with entropy coding | 10.0 | 500 | 8 | 14.0 | Limited Results |
| DPCM (delta coding with threshold) | 4.0 | 300 | 8 | -- | Sensitive (SF) Poor P Fidelity |
| DPCM Linear prediction | 2.5 | 250 | 12 | -- | High Fidelity |
| DPCM-Linear Prediction Interpolation, and Entropy coding | 7.8 | 500 | 8 | 3.5 | Sensitive (SF) and quantization |
| Orthogonal Transforms (CT, KLT,HT) | 3.0 | 250 | -- | -- | Lead-I |
| Dual application of K-L Transformation | 12.0 | 250 | 12 | -- | (X,Y,Z) Leads |
| Fourier Descriptors | 7.4 | 250 | 12 | 7.0 | (X,Y) Leads |

## 4.4 Transformation compression techniques

Unlike direct data compression, most of the transformation compression techniques have been employed in VCG or multilead ECG compression and require ECGwave detection. In general, transformation techniques involve preprocessing the input signal by means of a linear orthogonal transformation and properly encoding the transformed output (expansion coefficients) and reducing the amount of data needed to adequately represent the original signal. Upon signal reconstruction, an inverse transformation is performed and the original signal is recovered with a certain degree of error. However, the rationale is to efficiently represent a given data sequence by a set of transformation coefficients utilizing a series expansion (transform) technique.

Many discrete orthogonal transforms [28] [29] [30] have been employed in digital signal representation such as Karhunen-Loeve transform (KLT), Fourier (FT), Cosine (CT), Walsh (WT), Haar (HT), etc. The optimal transform is the KLT (also known as the principal components transform or the eignevector transform) in the sense that the least number of orthonormal functions is needed to represent the input signal for a given rms error. Moreover, the KLT results in decorrelated transform coefficients (diagonal covariance matrix) and minimizes the total entropy compared to any other transform. However, the computational time needed to calculate the KLT basis vectors (functions) is very intensive. This is due to the fact that the KLT basis vectors are based on determining the eigenvalues and corresponding eigenvectors of the covariance matrix of the original data, which can be a large symmetric matrix. The lengthy processing requirement of the KLT has led to the use of suboptimum transforms with fast algorithms (i.e., FT, WT, CT, HT, etc).

## 4.5 Conclusion

As discussed above direct data compression methods are widely employed in ECG data compression. This is mainly due to the ease of implementation of such techniques. On the other hand, limited work has been reported on ECG data compression by transformation techniques. This has been primarily due to the computational requirement, and in some cases due to the low achieved compression ratios (especially in single lead ECG compression).

ECG data compression techniques comparison, based in absolute terms on the reported compression ratios, is improper. In fact, the compression ratio calculation of such techniques has been based on comparing the resulted compression parameters with the number of samples in the original data [11]. Among many factors, sampling rate and precision of the "input" ECG data, and the word-length of the "output" compression parameters, which directly affect the compression ratio value, have not been taken into consideration.

Table 4.1 provides a summary of ECG data compression techniques in terms of compression ratio (CR), sampling frequency (SF) and A/D precision level, percent rms difference (PRD), and pertinent reported comments whenever available. The sampling rate and precision of ECG signals originally employed in each compression method are reported in an attempt to form some basis of comparison among such techniques.

We have made a broad discussion of the existing ECG data compression techniques in attempt to get comparison points for the technique to be developed. The next chapter deals with the proposed technique, which is artificial neural network based ECG compression system and performs experiments using real ECG data.

# CHAPTER FIVE

# THE PROPOSED TECHNIQUE

*ECG COMPRESSION SYSTEM BASED ON BACK- PROPAGATION ANN*

## 5.1 Introduction

BP neural network can be used to compress ECG signals because ECG signals are quasi-periodicity signals. The difference between cycles is little. When BP neural network is used to study each cycle of ECG waves, the little change between cycles can be expressed with hidden units' values, and weights don't change. So we can only store hidden units' values. If the number of hidden units is smaller than the number of input data, data are compressed. [31]

In this direct data compression technique the ECG data are first pre-processed. Four filters are adopted to process the data to improve its quality. Then the R-points of the ECG data are detected using a robust algorithm and the data are partitioned by taking R-points as base. The partitioned beats are then trained by a feed-forward artificial neural network and hidden values and output weights are saved.

## 5.2 The pre-processing of ECG data

Unfortunately, modern electrocardiographs record an ECG signal contaminated by various kinds of interference signals. These signals include the ubiquitous 60Hz powerline frequency, the depolarization of muscle tissue contraction creating electromyographic interference, and the rhythmic inhalation and exhalation during respiration causing a low frequency baseline drift. The ECG data can also be corrupted if the subject being tested moves suddenly, causing an abrupt baseline shifts.

The important compression parameters, compression ratio, compression precision, and compression rate are significantly affected by noise. Some noise sources can be eliminated before recording, such as powerline noise that can be easily removed with a 60Hz notch filter, but others, such as electromyographic interference and abrupt baseline shift, must be identified after the data has been collected due to their irregular nature. Thus various filters must be implemented to eliminate noise. These filters may include both analog and digital filters.

In this proposed work four digital filters are used to solve noise problem. These digital filters are simple, effective and real-time.

***Notch filter***

The filter is designed to get rid of the powerline interference [main paper] (In this dissertation, MIT-BIH database is used as a source of ECG data to test the algorithms. Hence powerline frequency is 50Hz). The algorithm of Notch tilter is:

$$y(n) = \frac{1}{2}[x(n) + x(n-k)] \tag{5.1}$$

***Low-pass filter***

The following Low-pass filter is used to eliminate low amplitude and high frequency noise [main paper].

$$y(n) = x(n) + 4x(n-1) + 6x(n-2) + 4x(n-3) + x(n-4) \tag{5.2}$$

The following low-pass filter is used to smooth the ECG waves.

$$y(n) = [x(n-1) + 2x(n) + x(n+1)]/4 \tag{5.3}$$

***High-pass filter***

The following high-pass filter is used to eliminate low frequency noise such as the drift of the baseline [31]. The algorithm is:

$$y(n) = 32x(n-16) - [y(n-1) + x(n) - x(n-32) \quad (5.4)$$

## 5.3 R-point detection

The ability to detect QRS complexes in ECG signals is a very important skill in the field of medical instrumentation. It is very important either in the identification of ECG waves or in the compression of ECG data.

In this work ECG data are first partitioned into different cycles before they are compressed. The partitioning is based on R points. The detection of R points is the precondition of the ECG data compression.

Numerous algorithms have been devised in order to accomplish this task of automated R point detection. Five algorithms described by Friesen et al [32] are discussed here. Three incorporated use of the amplitude and the first derivative of the ECG signal – they are designated as AF1, AF2 and AF3. The other two algorithms used only the first derivative and are designated as FD1 and FD2.

***Algorithms Based on Amplitude and First Derivative***

**AF1**: required finding three threshold constants, one for the ascending slope of the ECG signal, one for the descending slope, and one for the amplitude of the QRS complex. The algorithm calculates the first derivative (slope) of the ECG data is found by using the simplified derivative expression of an FIR filter (next sample minus previous sample). [32]

Using the first derivative the algorithm loops through the ECG sample and finds where three consecutive values in the slope vector are above the ascending threshold. Once it finds data that satisfies this condition, it looks for when two consecutive values that are below the descending slope threshold within 100 milliseconds, and if all data points between these ascending and descending points are above the amplitude threshold, the index of the first point above the ascending threshold is considered the location of the start of a QRS. This cycle is continued until the end of the sampled data is reached.

The algorithm based on AF1 is [32]:

$$Amplitude\ threshold = 0.3\,max[X(n)] \quad 1 < n < length(x)$$

$$Y(n) = X(n+1) - X(n-1) \quad 2 < n < length(x) - 1 \tag{5.5}$$

$$Y(i), Y(i+l), Y(i+2) > 0.5 \quad \text{and}$$

$$Y(j), Y(j+1) < -0.3 \quad \text{where} \quad (i+2) < j < (i+25)$$

and

$$X(i), X(i+l), \ldots, X(j+1) \geq amplitude\ threshold$$

**AF2**: utilizes its amplitude threshold differently than AF1. It first rectifies the ECG signal, and then sets any data point less than the amplitude threshold equal to the threshold value. The derivative of the resulting signal is found in the same way as in AF1, and any point found to have a slope above a threshold is considered a QRS complex [32]. The algorithm used based on AF2 is:

- A threshold is calculated as a fraction of the peak value of the ECG:

$$Amplitude\ threshold = 0.4 * max[X(n)]$$

- The raw data is then rectified:

$$Y0(n) = X(n) \qquad if\ X(n) >= 0 \qquad 1 < n < length(X)$$

$$Y0(n) = -X(n) \qquad if\ X(n) < 0 \qquad 1 < n < length(X)$$

- The rectified ECG is passed through a low level clipper:

$$Y1(n) = Y0(n) \qquad if\ Y0(n) >= amplitude\ threshold$$

$$Y1(n) = amplitude\ threshold \qquad if\ Y0(n) < amplitude\ threshold.$$

- The first derivative is calculated at each point of the clipped, rectified array:

$$Y2(n) = Y1(n+1) - Y1(n-1) \qquad 2 < n < length(y1)$$

- A QRS candidate occurs when a point in Y2( n ) exceeds the fixed constant threshold,

$$Y2(i) > 0.4 * max(Y2) \qquad (5.6)$$

**AF3**: computes the first derivative of the ECG signal (in the same way as AF1 and AF2), and then loops through the signal searching for four consecutive points whose derivatives exceed the pre-determined slope threshold. Once four consecutive points are found as such, a check is performed to see if the product of the first derivative and the amplitude of the next two sample points of the ECG are positive. If so, it identifies the first point exceeding the slope cutoff as a QRS location. [32]

The algorithm used based on AF3 is:

- The first derivative is calculated at each point of the ECG:

$$Y(n) = X(n+1) - X(n-1) \qquad 2 < n < length(x) - 1$$

- The first derivative array is then searched for points which exceed a constant threshold:

$$Y(i) \geq 0.15$$

- Then the next three derivative values $Y(i+1), Y(i+2), and\ Y(i+3)$ must also exceed 0.15.
- If the above conditions are met, point i can be classified as a QRS candidate if the next two sample points have positive slope amplitude products:

$$Y(i+1)X(i+1)\ and\ Y(i+2)X(i+2) > 0$$

***Algorithms Based on First Derivative Only***

**FD1**: only uses only the first derivate to find the QRS complexes. Unlike the other algorithms, it calculates the derivative using four data points – two before and two after the point in question. It then determines QRS locations by finding points whose derivatives exceed a set percentage of the maximum value of the derivative in the ECG. [32]

The algorithm used based on FD1 is:

- The first derivative is calculated for each point of the ECG, using Menard's formula:

$$Y(n) = -2X(n-2) - X(n-1) + x(n+1) + 2X(n+2)$$
$$3 < n < length(x) - 2$$

- The slope threshold is calculated as a fraction of the maximum slope for the first derivative array.

$$Slope\ threshold = 0.70\ max[Y(n)] \qquad 3 < n < length(x) - 2$$

- The first point that exceeds the slope threshold is taken as the onset of a QRS candidate:

$$Y(i) > slope\ threshold$$

**FD2**: uses only the first derivate, as calculated in the AF algorithms, to find the QRS complexes. It accomplishes this by finding one sample above the constant slope threshold value and then

checking the next three data points for a sample above the threshold. In other words, two out of four data points must have slopes exceeding a threshold in order to be identified as a QRS interval. The first point above the threshold is considered the start of the QRS. The rest of the peaks are found in the same fashion. [32]

The algorithm used based on FD2 is:

- The first derivative is calculated at each point of the ECG:

$$Y(n) = X(n+1) - X(n-1) \qquad 2 < n < length(x) - 1$$

- This array is searched until a point is found that exceeds the slope threshold:

$$Y(i) \geq 0.45$$

- A QRS candidate occurs if another point in the next three sample points also exceeds the threshold:

$$Y(i+1) > 0.45, or$$

$$Y(i+2) > 0.45, or$$

$$Y(i+3) > 0.45$$

***Selected algorithm***

For the ECG data used in this dissertation the second type of algorithm based on Amplitude and First Derivative (i.e. AF2) is used.

The performance of the selected QRS detection mechanism (i.e. AF2) is evaluated and summarized in table 5.1. The duration of the ECG data used is 1 min. and the sampling frequency is 360 Hz. Therefore the total number of points in a signal is 21,600.

**Table 5.1 Performance evaluation of the selected QRS detector**

| No. | ECG data Name | Original number of R points | Detected number of R points | False R points | Missed R points | Maximum R value | Remarks |
|---|---|---|---|---|---|---|---|
| 1 | Mba100 | 74 | 74 | 0 | 0 | 0.9 | |
| 2 | Mba101 | 71 | 72 | 1 | 0 | 1.44 | The false R point is a noised T point |
| 3 | Mba102 | 73 | 73 | 0 | 0 | 0.88 | |
| 4 | Mba103 | 70 | 70 | 0 | 0 | 1.79 | |
| 5 | Mba104 | 74 | 73 | 0 | 1 | 1.52 | |
| 6 | Mba105 | 83 | 83 | 0 | 0 | 1.74 | |
| 7 | Mba107 | 70 | 70 | 0 | 0 | 2.695 | |
| 8 | Mba111 | 69 | 69 | 0 | 0 | 1.085 | |
| 9 | Mba112 | 86 | 88 | 2 | 0 | 0.85 | |
| 10 | Mba114 | 54 | 54 | 0 | 0 | 0.69 | |
| 11 | Mba115 | 63 | 63 | 0 | 0 | 1.65 | |
| 12 | Mba117 | 50 | 50 | 0 | 0 | 1.86 | |
| 13 | Mba119 | 65 | 65 | 0 | 0 | 1.555 | |
| 14 | Mba121 | 60 | 60 | 0 | 0 | 1.475 | |
| 15 | Mba122 | 87 | 87 | 0 | 0 | 1.27 | |
| 16 | Mba123 | 49 | 49 | 0 | 0 | 1.55 | |
| 17 | Mba124 | 49 | 49 | 0 | 0 | 1.27 | |
| 18 | Mba201 | 90 | 90 | 0 | 0 | 0.94 | |
| 19 | Mba202 | 53 | 52 | 0 | 1 | 1.4 | One R point missed due to its abnormally low amplitude value |
| 20 | Mba205 | 89 | 89 | 0 | 0 | 0.835 | |
| 21 | Mba209 | 93 | 93 | 0 | 0 | 1.235 | |
| 22 | Mba210 | 91 | 85 | 0 | 6 | 1.045 | Six R point missed due to their very low amplitude value |
| 23 | Mba212 | 90 | 90 | 0 | 0 | 1.315 | |
| 24 | Mba214 | 77 | 74 | 0 | 3 | 1.995 | |
| 25 | Mba215 | 72 | 71 | 0 | 1 | 1.415 | |
| 26 | Mba219 | 74 | 74 | 0 | 0 | 1.6 | |
| 27 | Mba220 | 72 | 72 | 0 | 0 | 1.52 | |

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| 28 | Mba222 | 78 | 67 | 0 | 11 | 1.515 | R points missed due to their curved nature |
| 29 | Mba223 | 80 | 80 | 0 | 0 | 1.375 | |
| 30 | Mba228 | 69 | 69 | 0 | 0 | 2.26 | |
| 31 | Mba230 | 79 | 75 | 0 | 5 | 1.98 | R points missed due to their very low amplitude value |
| 32 | Mba231 | 63 | 63 | 0 | 0 | 1.52 | |
| 33 | Mba234 | 92 | 92 | 0 | 0 | 1.75 | |

As we can observe from the above table, the R-points of most of the ECG records are accurately detected. In record number 2 (i.e. Mba101) a single R-point is missed. This was due to a noised T point acting like an R-point.

In record numbers 5, 19, 22, 24, 25, 28, and 31 R-points are missed and in some of them in significant way. This will definitely create a problem in the compression system because the cycles corresponding to these R-points will not exist. One remedy to solve this problem is to use two QRS detection algorithms jointly and detect the R-points missed by one using the other.

But the proposed compression system has its own remedy too. If the some R-points are missed, it means there will be cycles which are unusually long, these cycles are filtered and will not be trained to the network and are left uncompressed.

Similarly if some false R-points are detected, there will be cycles which are unusually short, these cycles are selected and stored uncompressed. These will avoid any miss-training of the artificial neural network.

A sample of an ECG signal and the corresponding detected R waves is shown in figure 5.1.

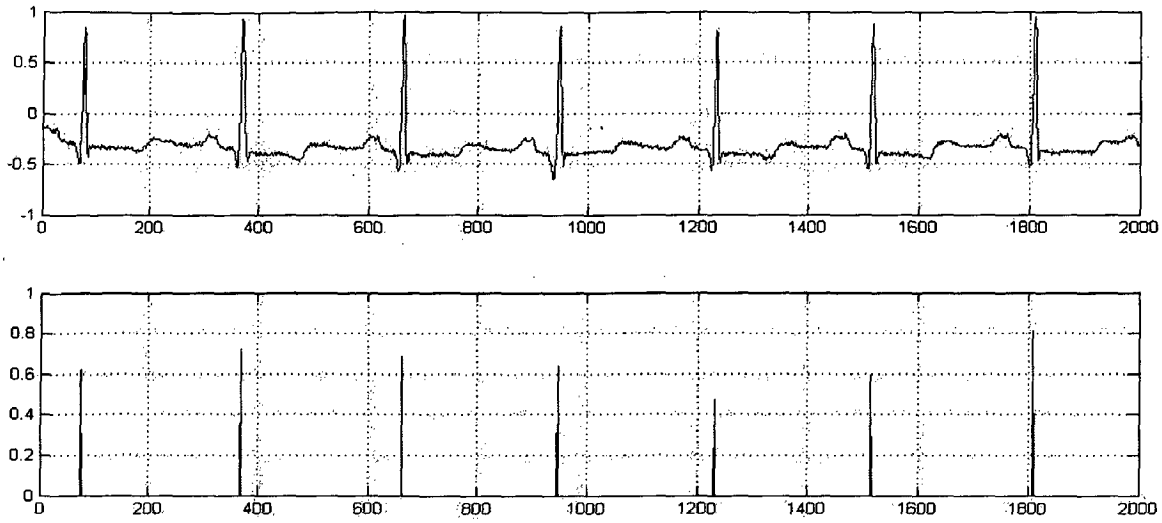



Figure 5.1 A sample of an ECG signal and the corresponding detected R waves

## 5.4 The compression system

A conceptual illustration of the compression system is shown in figure 5.9. The ECG data is fed to the system. The system reduces the dimension of the data and stores it or transmits it. The compressed data is represented by the hidden activation values. Whenever the data is required for display or processing, the compression system reconstructs it.

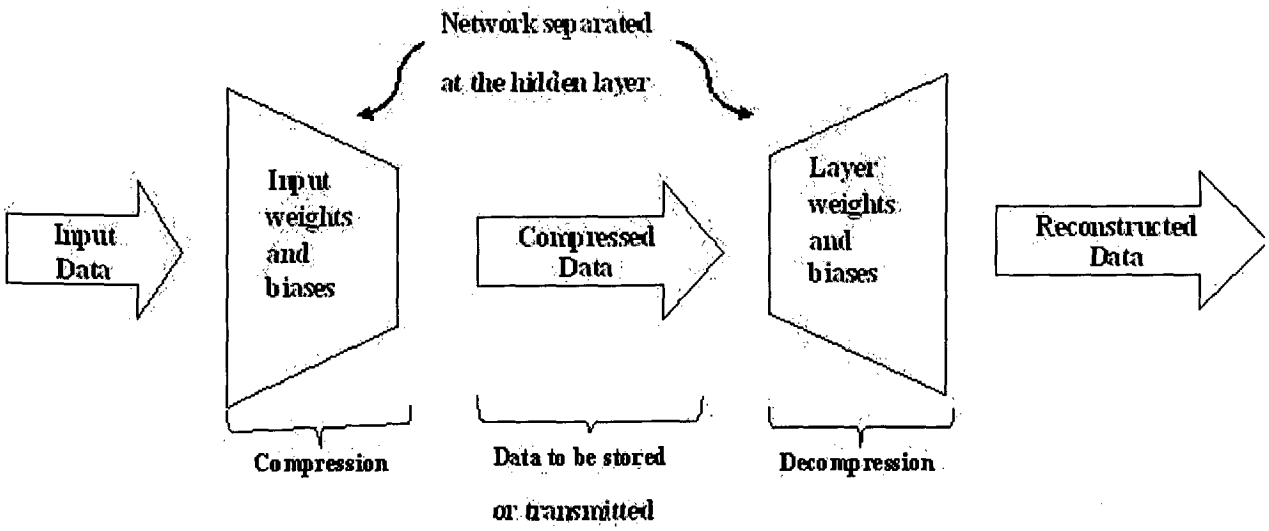



Figure 5.14 Illustration of the Compression System

### 5.4.1 Partitioning the ECG data

Before compressing an ECG data, it must be first partitioned in to cycles/beats. It is the cycles which are then trained to the network.

The partition of ECG cycles is based R points. We take the points among $d_1/3$ before an R point and $2d_2/3$ after the R point as one cycle (where $d_1$ is the space between the current R point and the previous R point and $d_2$ is the space between the current R point and the next R point). An R point's value is fixed on some input unit. [31]

If the point number of ECG cycles is different from the input unit number and the output unit number, we can process them differently according to the extent of the difference. If the difference is small and the number of ECG sampling points is smaller than the number of input units, we use suitable values to supply deficiencies of the input units. If the difference is small and the number of ECG sampling points is bigger than the number of input units, we take one point every other point in the foreparts and the tails of cycles and keep the number of input data to equal the number of input units.

### 5.4.2 Adopted network architecture

A three layer feed-forward BP neural network is adopted to compress ECG data (see fig 5.10). The number of the input neural units and the number of the output neural units equal the number of sampling points in an ECG cycle. The input unit number and the output unit number should be dynamic because heart rates are different for different persons or different time of the same person. We obtain the average heart rate of a patient through automatic measurement, and get the number of sampling points in an ECG cycle. The number is the input unit number and the output unit number.

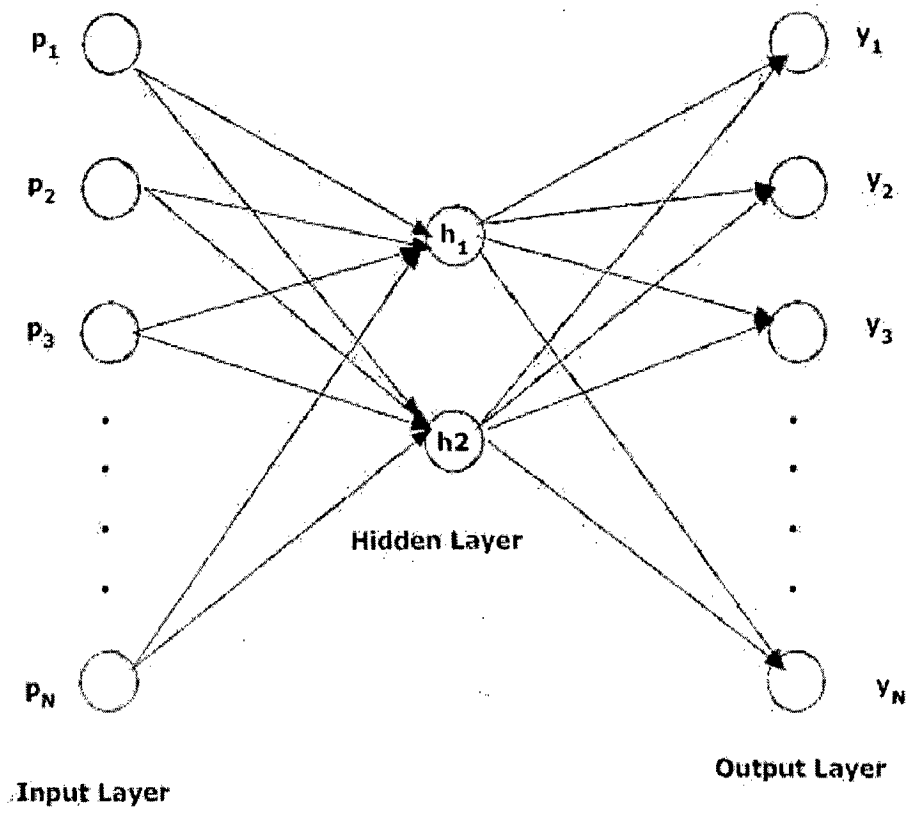



Figure5.3 A three layer feed-forward BP neural network

The neural unit number of hidden layer is 2. It is enough to express the difference of the same kind of ECG waves of the same person because there is little change between the ECG waves. Impermanent and highly aberrant ECG data are uncompressed and are directly stored. If the number of similar highly aberrant ECG data is five or more the system figures out new weights and hidden layer values.

### 5.4.3 Training the BP network

After the R-points of the ECG data are detected and the data is partitioned, it is trained to a 3-layer feed-forward network. Right after the network weights and biases are initialized, the network is ready to be trained for data compression. The training process requires a set of examples of proper network behavior—network inputs p and target outputs t.

During training the weights and biases of the network are iteratively adjusted to minimize the network performance, mean square error (mse) , which is the average squared error between the network outputs a and the target outputs t.

The training mode used in this work is batch training mode. The weights and biases of the network are updated only after the entire training set has been applied to the network. The gradients calculated at each training example are added together to determine the change in the weights and biases.

### 5.4.4 Selected BP training algorithm

The basic idea of the backpropagation learning algorithm is the repeated application of the chain rule to compute the influence of each weight in the network with respect to an arbitrary error function $E$:

$$\frac{\partial E}{\partial w_{ij}} = \frac{\partial E}{\partial s_i}\frac{\partial s_i}{\partial net_i}\frac{\partial net_i}{\partial w_{ij}} \tag{5.7}$$

where $w_{ij}$ is the weight from neuron $j$ to neuron $i$, $s_i$ is the output, and $net_i$ is the weighted sum of the inputs of neuron $i$. Once the partial derivative for each weight is known, the aim of minimizing the error function is achieved by performing a simple gradient descent:

$$w_{ij}(t+1) = w_{ij}(t) - \epsilon\frac{\partial E}{\partial w_{ij}}(t) \tag{5.8}$$

Obviously, the choice of the learning rate $\epsilon$, which scales the derivative, has an important effect on the time needed until convergence is reached. If it is set too small, too many steps are needed to reach an acceptable solution; on the contrary a large learning rate will possibly lead to oscillation, preventing the error to fall below a certain value.

What is often disregarded is that, the size of the actually taken weight-step $\Delta w_{ij}$ is not only dependent on the (adapted) learning-rate, but also on the partial derivative $\frac{\partial E}{\partial w_{ij}}$. So the effect of the carefully adapted learning-rate can be drastically disturbed by the unforeseeable behavior of the derivative itself. This was one of the reasons that lead to the development of a new learning scheme called Resilient backpropagation (RPROP): to avoid the problem of 'blurred adaptivity', RPROP changes the size of the weight-update $\Delta w_{ij}$ directly, i.e. without considering the size of the partial derivative.

***Resilient backpropagation***

Resilient propagation (RPROP), presented by Riedmiller and Braun [33], performs a direct adaptation of the weight step based on local gradient information. In crucial difference to some adaptation techniques, the effort of adaptation is not blurred by gradient behavior whatsoever.

To achieve this, we introduce for each weight its individual update-value $\Delta_{ij}$, which solely determines the size of the weight-update. This adaptive update-value evolves during the learning process based on its local sight on the error function *E*, according to the following learning-rule:

$$\Delta_{ij}^{(t)} = \begin{cases} \eta^+ * \Delta_{ij}^{(t-1)}, & if \ \frac{\partial E}{\partial w_{ij}}^{(t-1)} * \frac{\partial E}{\partial w_{ij}}^{(t)} > 0 \\ \eta^- * \Delta_{ij}^{(t-1)}, & if \ \frac{\partial E}{\partial w_{ij}}^{(t-1)} * \frac{\partial E}{\partial w_{ij}}^{(t)} < 0 \\ \Delta_{ij}^{(t-1)}, & else \end{cases} \tag{5.9}$$

Where $0 < \eta^- < 1 < \eta^+$

Verbalized, the adaptation-rule works as follows: Every time the partial derivative of the corresponding weight $w_{ij}$ changes its sign, which indicates that the last update was too big and the algorithm has jumped over a local minimum, the update-value $\Delta_{ij}$ is decreased by the factor $\eta^-$. If the derivative retains its sign, the update-value is slightly increased in order to accelerate convergence in shallow regions.

Once the update-value for each weight is adapted, the weight-update itself follows a very simple rule: if the derivative is positive (increasing error), the weight is decreased by its update-value, if the derivative is negative, the update-value is added:

$$\Delta w_{ij}^{(t)} = \begin{cases} -\Delta_{ij}^{(t)}, & if \ \frac{\partial E}{\partial w_{ij}}^{(t)} > 0 \\ +\Delta_{ij}^{(t)}, & if \ \frac{\partial E}{\partial w_{ij}}^{(t)} < 0 \\ 0, & else \end{cases} \tag{5.10}$$

$$w_{ij}^{(t+1)} = w_{ij}^{(t)} + \Delta w_{ij}^{(t)}$$

The following pseudo-code fragment shows the kernel of the RPROP adaptation and learning process. The **minimum** (**maximum**) operator is supposed to deliver the minimum (maximum) of two numbers; the sign operator returns +1, if the argument is positive, -1, if the argument is negative, and 0 otherwise.

At the beginning all update values $\Delta_{ij}$ are set to a reasonably chosen initial value $\Delta_0$. The initialization and assignment of all the parameters is discussed in [33].

*For all weights and biases* {

$$\textbf{if}\left(\frac{\partial E}{\partial w_{ij}}(t-1)*\frac{\partial E}{\partial w_{ij}}(t)>0\right)\textbf{then}\{$$

$$\Delta_{ij}(t)=\textbf{minimum}(\Delta_{ij}(t-1)*\eta^{+},\Delta_{max})$$

$$\Delta w_{ij}(t)=-\textbf{sign}\left(\frac{\partial E}{\partial w_{ij}}(t)\right)*\Delta_{ij}(t)$$

$$w_{ij}(t+1)=w_{ij}(t)+\Delta w_{ij}(t)$$

}

$$\textbf{elseif}\left(\frac{\partial E}{\partial w_{ij}}(t-1)*\frac{\partial E}{\partial w_{ij}}(t)<0\right)\textbf{then}\{$$

$$\Delta_{ij}(t)=\textbf{maximum}(\Delta_{ij}(t-1)*\eta^{-},\Delta_{min})$$

$$w_{ij}(t+1)=w_{ij}(t)-\Delta w_{ij}(t-1)$$

$$\frac{\partial E}{\partial w_{ij}}(t)=0$$

}

$$\textbf{elseif}\left(\frac{\partial E}{\partial w_{ij}}(t-1)*\frac{\partial E}{\partial w_{ij}}(t)=0\right)\textbf{then}\{$$

$$\Delta w_{ij}(t)=-\textbf{sign}\left(\frac{\partial E}{\partial w_{ij}}(t)\right)*\Delta_{ij}(t)$$

$$w_{ij}(t+1)=w_{ij}(t)+\Delta w_{ij}(t)$$

}

}

### 5.4.5 Performance measuring techniques

Other than the network performance function (i.e. the mean square error-mse), the original and reconstructed ECG data are analyzed and the network performance is investigated using techniques described below.

- **PRD:** percent root-mean-square difference (PRD calculation is as follows:

$$PRD = \sqrt{\frac{\sum_{i=1}^{n}[x_{org}(i) - x_{rec}(i)]^2}{\sum_{i=1}^{n} x_{org}{}^2(i)}} * 100 \quad (5.11)$$

where $x_{org}$ and $x_{rec}$ are samples of the original and reconstructed data sequences.

- **Compression ratio:** is defined as the ratio of the size of the original ECG data( to those data the network is applied) to the size of the compressed ECG data.
- **Percent RR-interval error:** is the percentage error between the values of the R-R intervals of the original ECG data and the Corresponding values of the R-R intervals of the reconstructed ECG data.
- **Reconstruction error (Percent data lost):** is the percentage data lost (or gained) during compression.
- **Onset of P-wave error**: indicates how the onsets of P-waves of the original ECG data are represented in the reconstructed ECG data. It is the percent root-mean-square difference (PRD) of the two onsets.
- **Offset of T-wave error:** indicates how the offsets of T-waves of the original ECG data are represented in the reconstructed ECG data. It is the percent root-mean-square difference(PRD) of the two offsets
- Visual inspection: The features of each ECG component discussed in chapter 2 are visually inspected. The offsets of P-waves and onsets of T-waves are especially examined. Changes to these components are noted if they are caused by the compression system.

N.B. The onset and offset of P-wave and T-wave are defined as the points at which the P-wave and T-wav exceed and return to the baseline (isoelectric line), respectively. Since these points are important in ECG interpretation, they have to be included as evaluation parameters of the proposed compression system.

### 5.4.6 Reconstruction

During reconstruction the data must be figured out according to the way they are saved. If the data are uncompressed they are saved as they are. If the data are compressed the reconstructed data is figured out according to formula 5.6.

$$out\_ecg_k = \sum_{j=1}^{HL} w_{kj}\, h_j \qquad k = 1{:}OL \tag{5.6}$$

Where $OL$ = output layer size

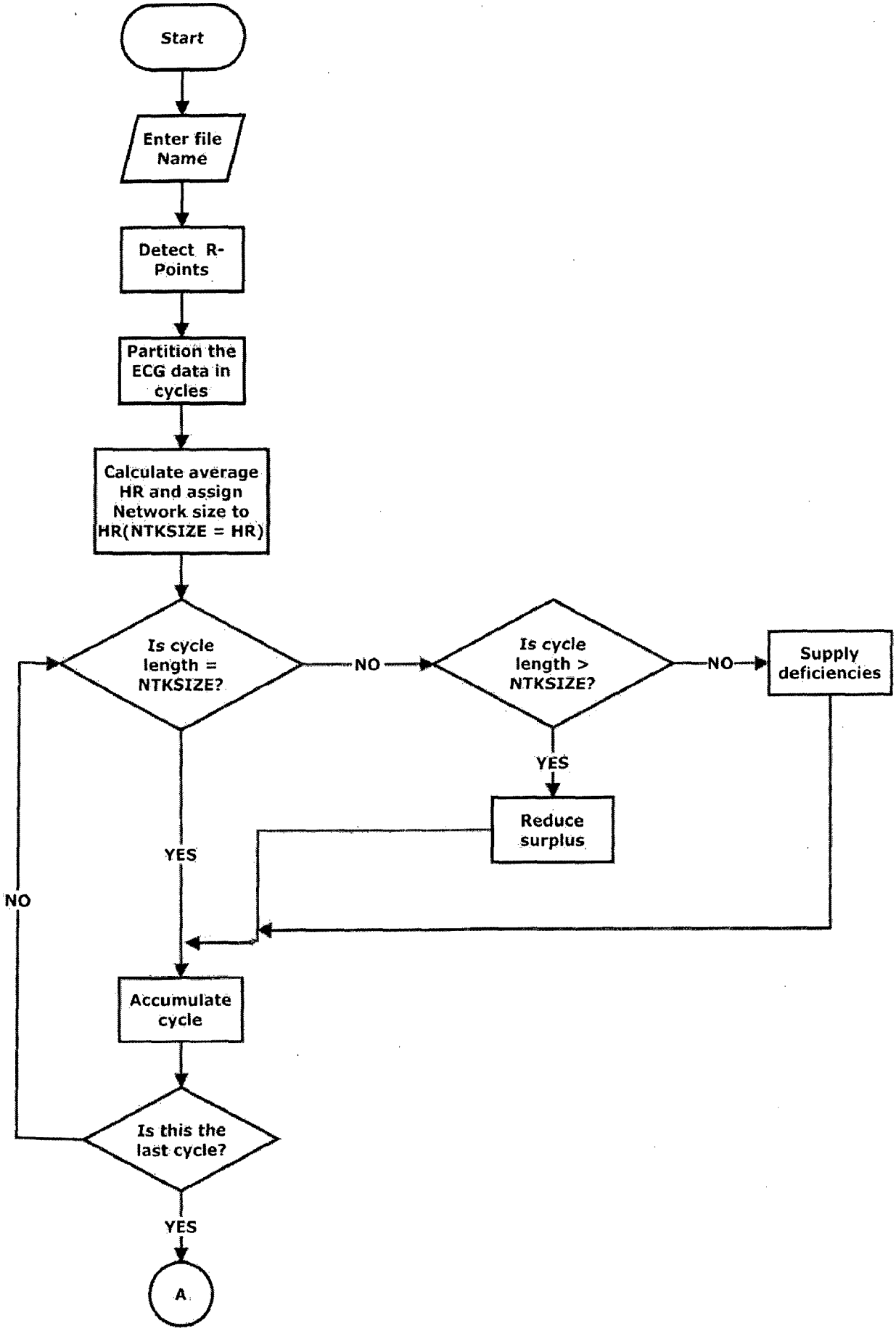
HL = hidden layer size

W = out put layer weight

$h$ = hidden layer activation

The R-peak value of each cycle of the reconstructed ECG data is replaced by the corresponding cycle R-peak value of the original ECG data. This is important to avoid any R-peak error and therefore the error is nil.

The overall compression system can be summarized with flow charts given in figure 5.11 and figure 5.12. The flow chart in figure 5.11 shows the compression stage and the flow chart in figure 5.12 shows the reconstruction stage.


Start
Enter file Name
Detect R-Points
Partition the ECG data in cycles
Calculate average HR and assign Network size to HR(NTKSIZE = HR)
Is cycle length = NTKSIZE?
NO
Is cycle length > NTKSIZE?
NO
Supply deficiencies
YES
Reduce surplus
YES
NO
Accumulate cycle
Is this the last cycle?
YES
A

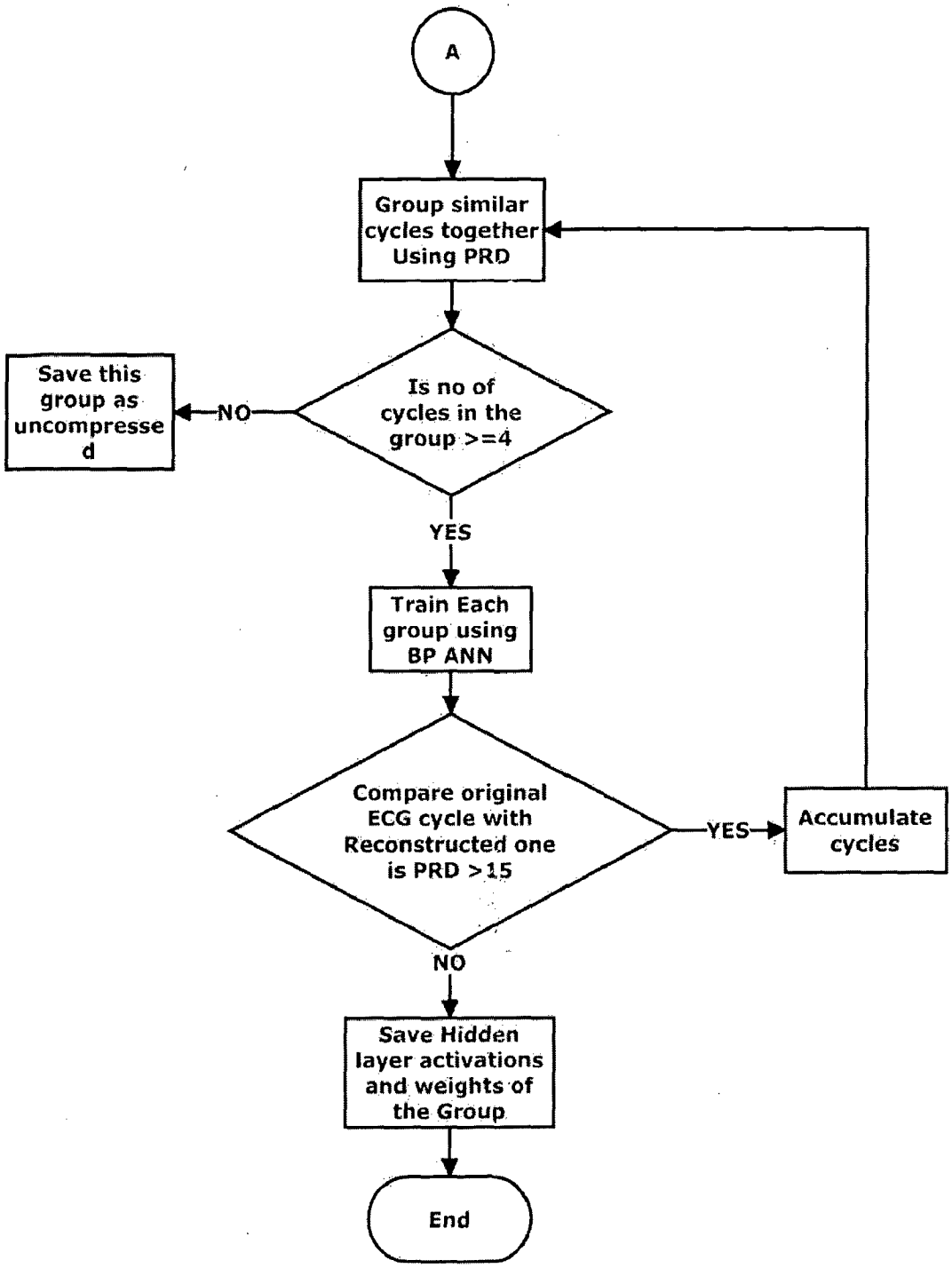



Figure5.4 Flow chart of the compression stage.

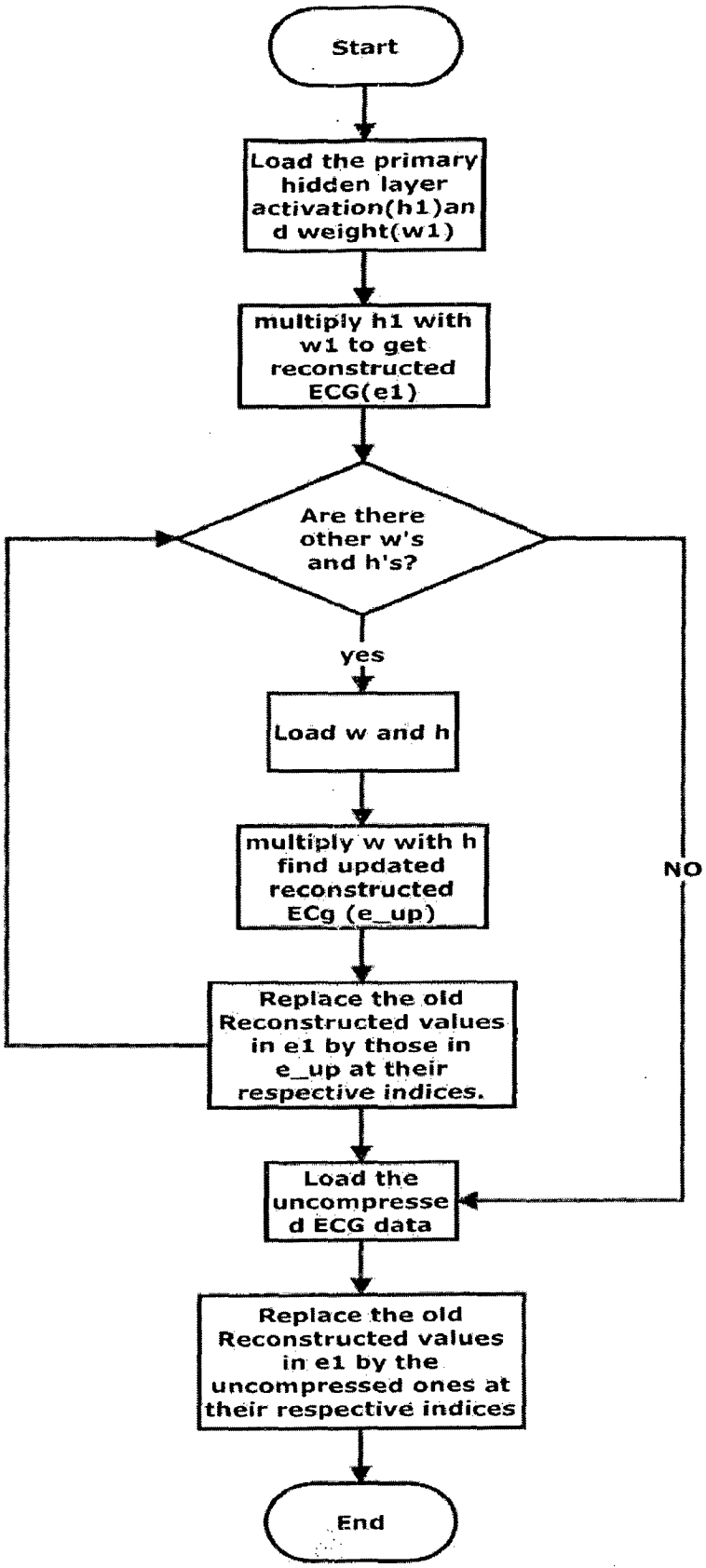



Figure5.5 Flow chart of the reconstruction stage.

# CHAPTER SIX

# RESULT AND DISCUSSION

## 6.1 Description of the ECG data used

The MIT-BIH Arrhythmia Database contains 48 half-hour excerpts of two-channel ambulatory ECG recordings, obtained from 47 subjects studied by the BIH Arrhythmia Laboratory between 1975 and 1979. Twenty-three recordings were chosen at random from a set of 4000 24-hour ambulatory ECG recordings collected from a mixed population of inpatients (about 60%) and outpatients (about 40%) at Boston's Beth Israel Hospital; the remaining 25 recordings were selected from the same set to include less common but clinically significant arrhythmias that would not be well-represented in a small random sample. [34]

The recordings were digitized at 360 samples per second per channel with 11-bit resolution over a 10 mV range.

Out of the 48 half-hour ECG records, 33 records are selected for experimental testing of the artificial neural network based ECG compression system. Every selected record contains a 1 minute long data. The sampling frequency is 360, so each record includes 21,600 sample points. The naming of each record is done by adding a prefix 'mba-'on the original numeric name found on the database (e.g. mba100).

## 6.2 Experimenting with MIT-BIH arrhythmia database

Table 6.1 shows the experimental result of each record and displays the compression ratio, compression precision and compression time of each experiment along with other performance testing parameters.

Table 6.1 Experimental result of ANN based ECG compression

| No. | ECG data | Network CR | Overall CR | Compr-ession time(s) | Compression precisions(PRD) | | | Average R-R interval error (%) | Data loss (%) |
|---|---|---|---|---|---|---|---|---|---|
| | | | | | overall precision | Network precision | Minimum precision | | |
| 1 | cmba100 | 10.06 | 4.25 | 39 | 6.95 | 8.83 | 3.13 | 4.00 | 3.11 |
| 2 | cmba101 | 5.51 | 2.16 | 40 | 7.90 | 9.30 | 2.37 | 5.87 | -1.31 |
| 3 | cmba102 | 5.59 | 2.36 | 43 | 8.43 | 8.43 | 6.16 | 4.31 | 0.26 |
| 4 | cmba103 | 5.90 | 2.82 | 39 | 7.01 | 9.61 | 1.67 | 5.16 | -3.44 |
| 5 | cmba104 | 10.12 | 1.58 | 36 | 5.05 | 10.4 | 5.49 | 4.54 | 3.45 |
| 6 | cmba105 | 20.20 | 4.1 | 37 | 6.65 | 8.05 | 4.31 | 4.29 | 2.11 |
| 7 | cmba107 | 4.66 | 2.00 | 40 | 8.64 | 9.98 | 4.32 | 4.14 | -2.64 |
| 8 | cmba111 | 3.61 | 1.90 | 47 | 8.16 | 12.1 | 1.79 | 4.38 | 3.96 |
| 9 | cmba112 | 24.13 | 11.7 | 39 | 4.14 | 5.39 | 2.00 | 4.45 | -1.31 |
| 10 | cmba114 | 3.06 | 1.85 | 32 | 7.55 | 9.65 | 2.02 | 3.98 | -0.26 |
| 11 | cmba115 | 2.92 | 1.67 | 37 | 5.16 | 8.16 | 2.91 | 4.33 | -3.95 |
| 12 | cmba117 | 15.26 | 5.87 | 26 | 6.52 | 7.22 | 3.39 | 3.04 | 0.03 |
| 13 | cmba119 | 2.95 | 2.00 | 34 | 7.05 | 10.5 | 4.11 | 3.33 | -2.63 |
| 14 | cmba121 | 18.10 | 6.93 | 30 | 5.73 | 6.03 | 2.04 | 9.03 | 0.02 |
| 15 | cmba122 | 25.6 | 10.0 | 39 | 5.33 | 5.90 | 2.61 | 8.46 | 1.84 |
| 16 | cmba123 | 2.50 | 1.85 | 24 | 9.16 | 13.6 | 4.10 | 3.58 | 2.37 |
| 17 | cmba124 | 14.26 | 5.08 | 26 | 8.73 | 8.73 | 3.68 | 4.70 | 0.79 |
| 18 | cmba201 | 3.91 | 1.80 | 51 | 4.46 | 10.9 | 5.84 | 9.09 | 3.95 |
| 19 | cmba202 | 4.39 | 1.69 | 30 | 5.64 | 10.6 | 2.22 | 4.55 | -3.44 |
| 20 | cmba205 | 25.35 | 10.0 | 40 | 5.54 | 6.04 | 3.71 | 4.78 | 1.05 |
| 21 | cmba209 | 6.90 | 2.88 | 46 | 9.44 | 11.6 | 5.63 | 3.98 | -1.32 |
| 22 | cmba212 | 3.11 | 1.5 | 45 | 6.33 | 11.3 | 3.58 | 5.62 | -2.38 |
| 23 | cmba213 | 28.58 | 7.9 | 46 | 8.67 | 9.72 | 3.83 | 0.71 | -0.26 |
| 24 | cmba214 | 6.62 | 3.47 | 41 | 8.00 | 9.98 | 4.75 | 10.01 | 0.25 |
| 25 | cmba217 | 4.73 | 2.84 | 39 | 8.17 | 10.7 | 1.63 | 5.26 | 1.05 |
| 26 | cmba219 | 4.27 | 2.19 | 44 | 4.21 | 8.79 | 0.31 | 9.85 | 6.06 |
| 27 | cmba220 | 7.20 | 4.67 | 41 | 6.30 | 6.97 | 1.56 | 5.23 | 2.63 |
| 28 | cmba222 | 5.73 | 1.24 | 31 | 1.67 | 9.1 | 6.74 | 10.31 | 3.95 |
| 29 | cmba223 | 22.01 | 5.84 | 36 | 6.43 | 7.63 | 3.71 | 3.79 | -1.31 |
| 30 | cmba230 | 3.25 | 1.84 | 50 | 7.18 | 10.1 | 1.36 | 6.01 | 4.22 |
| 31 | cmba231 | 3.31 | 1.95 | 41 | 5.25 | 9.25 | 3.90 | 8.18 | 5.01 |
| 32 | cmba234 | 12.22 | 4.98 | 42 | 8.22 | 9.2 | 4.25 | 2.28 | 0.79 |
| 33 | cmba116 | 22.93 | 7.58 | 36 | 7.21 | 8.21 | 3.71 | 3.04 | 0.78 |
| **Average** | | **10.27** | **3.95** | **38.39** | **6.69** | **8.30** | **3.41** | **5.31** | **2.17** |

**Compression ratio (CR)**

Two kinds of compression ratios are presented in table 6.1, the Network compression ratio and Overall compression ratio. The reason of presenting two kinds of compression ratios is due to the uncompressed ECG data. The Network compression ratio excludes the uncompressed ECG data and examines only the performance of the BP ANN. But the Overall compression ratio calculates the compression performance of the whole system (i.e including the uncompressed ECG data). When the uncompressed ECG data are taken in to consideration, it is obvious that the compression ratio degrades.

Taking the Network alone, a compression ratio as high as 28:1 is reached and an average compression ratio (of all the 33 records used) of 10:1 is achieved. This shows how ANN can be powerful in compressing ECG data.

Regarding the Overall compression system, the maximum compression ratio achieved is 12:1. The average compression ratio in this case has degraded to 4:1. This implies many ECG beats are stored uncompressed due to their impermanent and highly aberrant nature. But in practical case, ECG data are likely to have little aberrance and impermanence per record, especially when the ECG data is long. Hence it is well fair to take the average compression ratio of the network as the compression ratio of the system. (see table 6.2 for more justification)

Table 6.2 shows how the Network and Overall compression systems are improved as more ECG beats are used. A 5 minute long data of 10 ECG records from the above 33 are utilized to test the compression system.

Table6.8 Testing the compression system for 5min. ECG data

| No. | ECG data | New Network CR | Old Network CR (i.e. 1 min) | New Overall CR | Old Overall CR (i.e. 1 min) |
|---|---|---|---|---|---|
| 1 | 5min_cmba100 | 21.28 | 10.06 | 7.97 | 4.25 |
| 2 | 5min_cmba101 | 21.23 | 5.51 | 7.84 | 2.16 |
| 3 | 5min_cmba103 | 15.90 | 5.90 | 5.31 | 4.1 |
| 4 | 5min_cmba105 | 20.38 | 20.20 | 4.78 | 2.82 |
| 5 | 5min_cmba112 | 39.07 | 24.13 | 13.9 | 11.7 |
| 6 | 5min_cmba114 | 4.36 | 3.06 | 2.05 | 1.85 |
| 7 | 5min_cmba115 | 7.28 | 2.92 | 3.17 | 1.67 |
| 8 | 5min_cmba117 | 30.77 | 15.26 | 7.57 | 5.87 |
| 9 | 5min_cmba121 | 32.6 | 18.10 | 7.48 | 6.93 |
| 10 | 5min_cmba205 | 25.35 | 25.6 | 10.0 | 7.58 |

Table 6.2 clearly shows that the longer the ECG data is the more improved the Network and Overall compression ratios will be. This indicates that the proposed compression system is best suitable for Holter or ambulatory ECG recording system.

**Compression precision**

Akin to to the compression ratio, the compression precision is also presented in two categories. One is the Network compression precision and the other is Overall compression precision. In a similar manner the Overall compression precision includes the uncompressed ECG data and the Network compression precision excludes them. One may argue that why would the uncompressed ECG data be considered to calculate the overall compression precision, if they are not touched by the network at all. My response would be; the uncompressed ECG data are those who are different from the other beats and which are highly aberrant. This beats are selected in a smart way and are made to bypass the network intentionally and stored in their original condition. This will avoid the wrong "prediction" of these beats. This deliberate action is

a virtue of the proposed compression system, and hence those uncompressed beats should be included in the calculation of the overall compression precision of the system.

Both the Network and Overall compression precisions are calculated in terms of the Percent Root-mean-square Difference (PRD) of the original ECG data and the reconstructed ECG data, in beat by beat manner.

Table 6.1 has displays the Overall compression precision, the Network compression precision and the Minimum compression precision for all the 33 records used in the experiment. The average Overall compression precision for the total of the 33 records is 6.69. The average Network precision is a little greater, it is 8.30. In both cases, the average precision is less than 10% which indicates that more than 90% of the original ECG data is reconstructed successfully. The minimum compression precision achieved by the compression system is 0.31.

**Compression time**

As to the compression time, for 1 minute long records the maximum time required is 50 seconds and a compression time as low as 24s is reached. This implies the proposed compression system is real-time.

**R-R interval error**

Some ECG arrhythmias, such as sinus arrhythmia, have a considerable R-R interval variation among successive ECG beats. This variation should be preserved during compression. Therefore the average percentage R-R interval difference between the original ECG record and the reconstructed ECG record is calculated and is included as performance evaluator. As table 6.1 shows, the average R-R interval error is 5%, which is acceptable for most arrhythmia types.

**Percentage data loss**

The last column of table 6.1 displays the percentage data loss after reconstruction. Since the compression system is lossy compression some data could be lost or gained (information lost). In this compression system not much loss of data is observed. The average loss of data of all experiments (i.e. 33 records) is 0.71%. It is very low to be significant.

**Onset and Offset of P-wave and T-wave**

Ten sample ECG data are selected to test how the onsets and offsets of the P-waves and the T-waves are affected by the proposed compression system. The result is presented in table 6.2 below.

Table6.9 Compression system effect on onsets and offsets of P-waves and T-waves

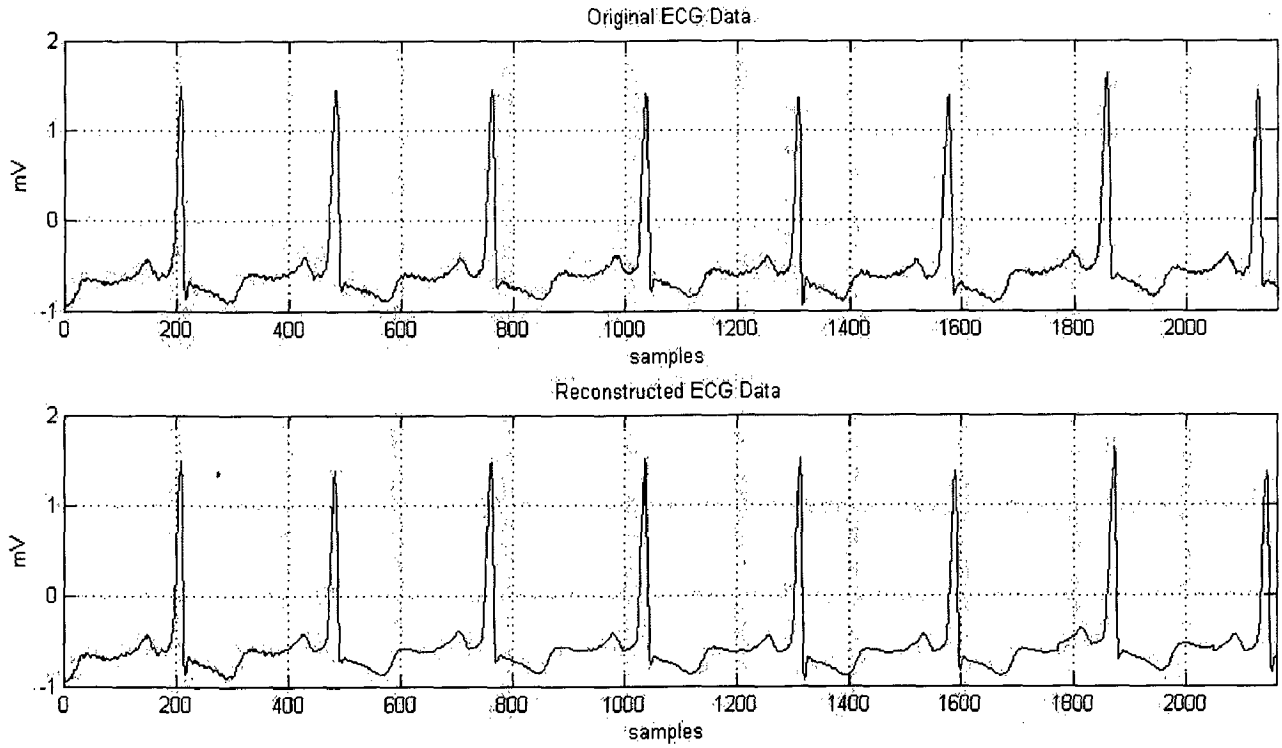
| No. | ECG record | Average error of onset of P-wave (PRD) | Visual inspection of offset of P-wave | Average error of offset of T-wave (PRD) | Visual inspection of onset of T-wave |
|---|---|---|---|---|---|
| 1 | cmba116 | 7.52 | Well represented | 9.53 | Well represented |
| 2 | cmba223 | 5.74 | Well represented | 8.97 | Well represented |
| 3 | cmba213 | 10.76 | Fairly represented | 16.62 | Fairly represented |
| 4 | cmba124 | 8.78 | Well represented | 14.9 | Fairly represented |
| 5 | cmba121 | 6.51 | Well represented | 12.02 | Fairly represented |
| 6 | cmba122 | 4.94 | Well represented | 7.27 | Well represented |
| 7 | cmba112 | 4.34 | Well represented | 7.35 | Well represented |
| 8 | cmba105 | 21.36 | Poorly represented | 20.54 | Poorly represented |
| 9 | cmba100 | 7.69 | Well represented | 12.62 | Fairly represented |
| 10 | cmba102 | 13.49 | Fairly represented | 14.15 | Fairly represented |

As we can observe from Table 6.2, most of the onsets and offsets of P-wave and T-wave are well represented or fairly represented. Most of the ECG data conceded an average error less than 10%. This result indicates that the proposed compression system is reliable.

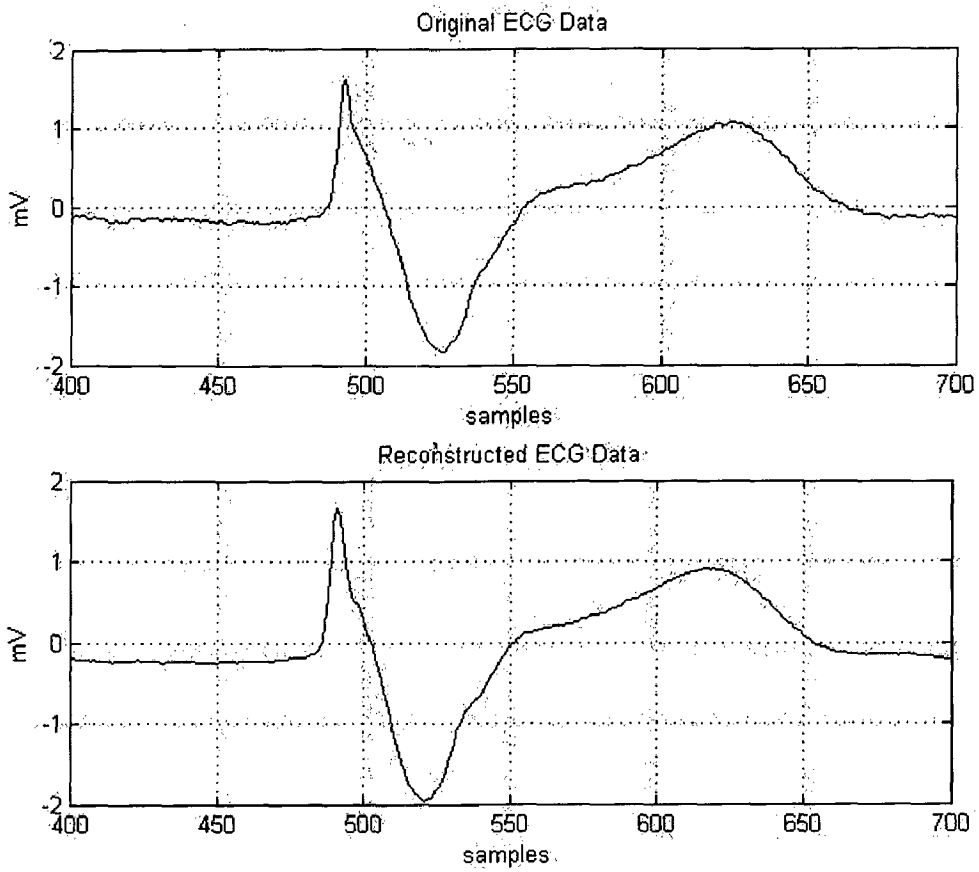
**Visual Inspection**

Most importantly, a thorough visual inspection is done on each experiment and the reconstructed ECG data are compared with the original ones. Even though few irregular extensions and shortenings of the Q-wave and the S-wave are observed, since they occur in small numbers and at low scale, we can conclude that the proposed compression system is significantly efficient.

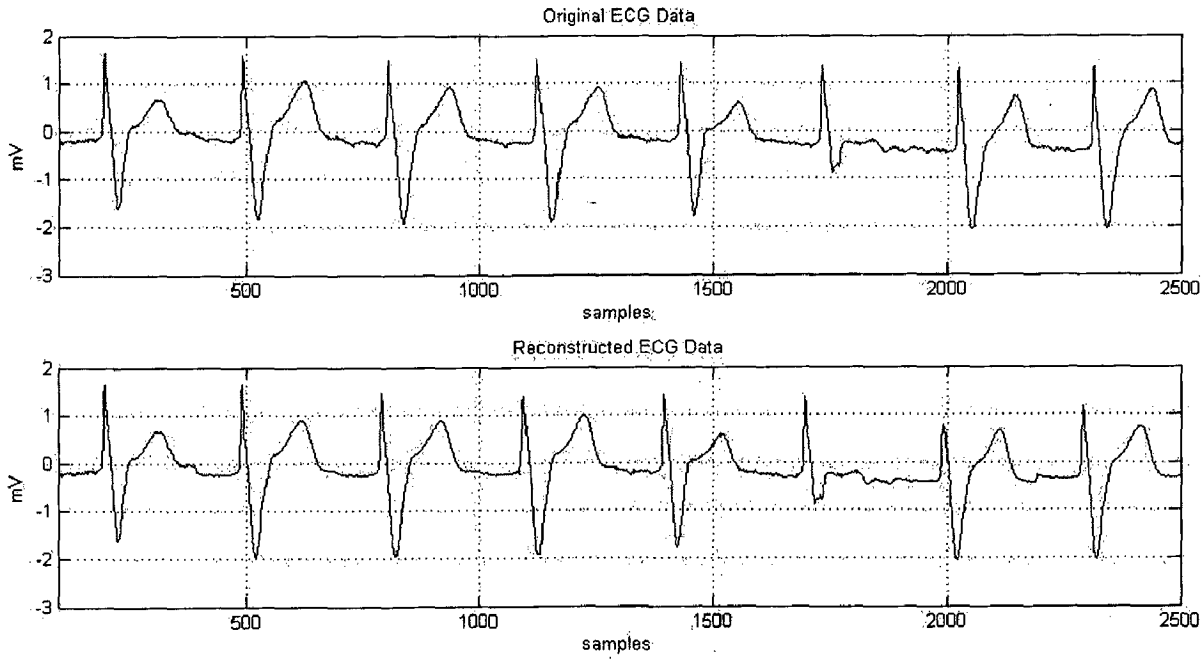
Sample test result are shown in figure 6.1. The figures show the ECG data before compression and after compression.




a) 6-seconds excerpt of the original and reconstructed data




b) A closer look at the original and reconstructed ECG data




c) A morphologically unique beat uncompressed successfully

**Figure 6.1 Sample test results of the compression system**

## 6.3 Effect of neural unit number of hidden layer on the compression system performance

At the beginning of my work I have made an assumption that the difference between ECG beats of the same record is little and therefore I have limited the hidden layer unit number to 2. All the experiments are done using this setup. In this section few experiments are done using different values of the neural unit number of the hidden layer.

Three records, cmba100, cmba112 and cmba217 are selected for the experiment and hidden layer neural unit numbers of 2, 4, 8, 12, 24, 36, and 72 are used. The results are shown in table 6.4, 6.5 and 6.6. The Network compression ratio and Network compression precision of all the three records are plotted versus the hidden layer size in figure 6.2, 6.3 and 6.4.

As we can observe it from the figures, when the hidden layer size increases, the network compression ratio decreases exponentially while the network compression precision shows little improvement. Hence the selection of hidden layer is a trade off between the compression ratio and the compression precision.

**Table6.10 Experimenting hidden layer size Effect using record cmba100**

| Hidden Layer size | Network CR | Overall CR | Compression time (sec) | Compression precisions(PRD) | | |
|---|---|---|---|---|---|---|
| | | | | overall precision | Network precision | Minimum precision |
| 2 | 10.06 | 4.25 | 39 | 7.17 | 8.56 | 3.11 |
| 4 | 6.11 | 3.6 | 37 | 6.84 | 8.03 | 2.87 |
| 8 | 3.64 | 2.87 | 38 | 6.53 | 7.32 | 2.10 |
| 12 | 2.56 | 2.20 | 39 | 5.87 | 6.58 | 0.41 |
| 18 | 1.66 | 1.53 | 41 | 6.25 | 7.34 | 1.91 |
| 24 | 1.31 | 1.26 | 44 | 5.95 | 6.78 | 1.55 |
| 36 | 0.87 | 0.88 | 41 | 6.37 | 7.36 | 2.20 |
| 72 | 0.43 | 0.47 | 54 | 5.29 | 6.21 | 1.44 |

**Table6.11 Experimenting hidden layer size Effect using record cmba112**

| Hidden Layer size | Network CR | Overall CR | Compression time (sec) | Compression precisions(PRD) | | |
|---|---|---|---|---|---|---|
| | | | | overall precision | Network precision | Minimum precision |
| 2 | 21.53 | 11.7 | 39 | 3.88 | 4.06 | 2.01 |
| 4 | 13.23 | 8.87 | 36 | 3.17 | 3.33 | 1.45 |
| 8 | 7.45 | 5.92 | 39 | 2.83 | 2.97 | 1.76 |
| 12 | 5.19 | 4.44 | 37 | 3.24 | 3.40 | 1.95 |
| 18 | 3.6 | 3.23 | 37 | 3.57 | 3.74 | 2.00 |
| 24 | 2.73 | 2.54 | 40 | 2.36 | 2.48 | 1.82 |
| 36 | 1.84 | 1.78 | 41 | 2.88 | 3.02 | 1.76 |
| 72 | 0.94 | 0.93 | 55 | 2.70 | 2.83 | 1.72 |

**Table6.12 Experimenting hidden layer size Effect using record cmba217**

| Hidden Layer size | Network CR | Overall CR | Compression time (sec) | Compression precisions(PRD) | | |
|---|---|---|---|---|---|---|
| | | | | overall precision | Network precision | Minimum precision |
| 2 | 5.79 | 2.73 | 39 | 7.99 | 10.50 | 1.49 |
| 4 | 9.67 | 3.08 | 32 | 7.05 | 9.44 | 4.14 |
| 8 | 3.18 | 2.26 | 37 | 6.76 | 8.28 | 1.41 |
| 12 | 2.15 | 1.73 | 39 | 6.40 | 8.12 | 2.21 |
| 18 | 1.48 | 1.34 | 39 | 6.47 | 8.21 | 1.32 |
| 24 | 1.12 | 1.09 | 44 | 6.47 | 8.21 | 1.69 |
| 36 | 0.71 | 0.77 | 39 | 5.87 | 8.01 | 1.00 |
| 72 | 0.35 | 0.43 | 56 | 5.26 | 7.32 | 0.80 |

But despite the little improvements of the compression precisions with the increase in hidden layer size, a worth tradable value is not attained before entering the warning(yellow) and danger(red) zone. Hence for this proposed compression system, a hidden layer neural unit number of 2 is enough to compromise the two important parameters, compression ratio and compression precision.

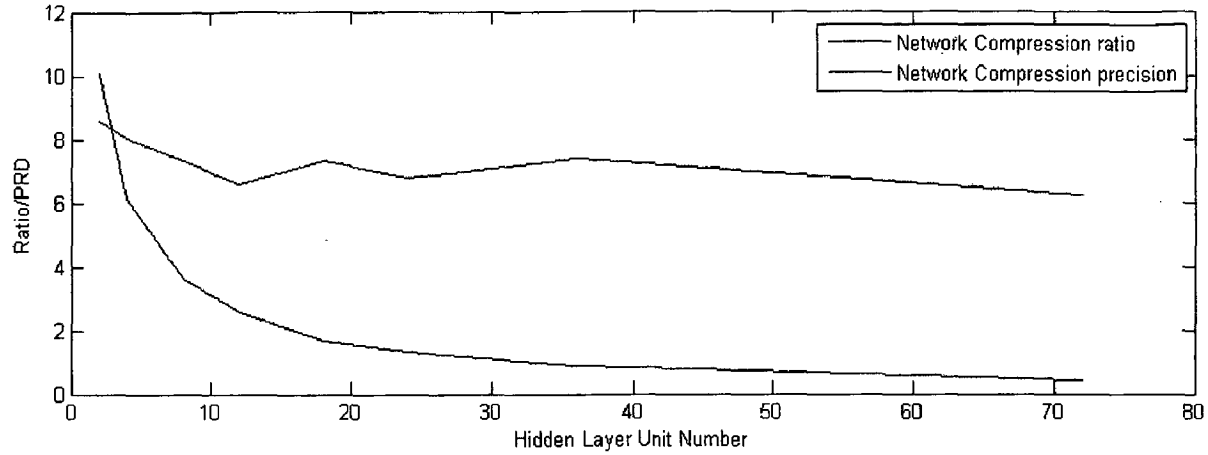



**Figure6.2 Effect of hidden layer size on PRD and comp. ratio / record cmba100**

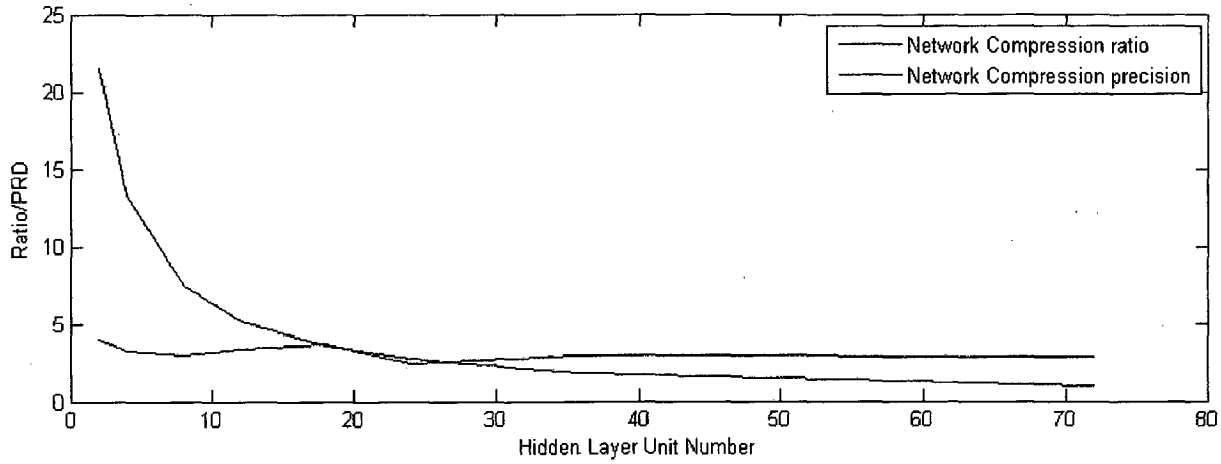



**Figure6.3 Effect of hidden layer size on PRD and comp. ratio / record cmba112**

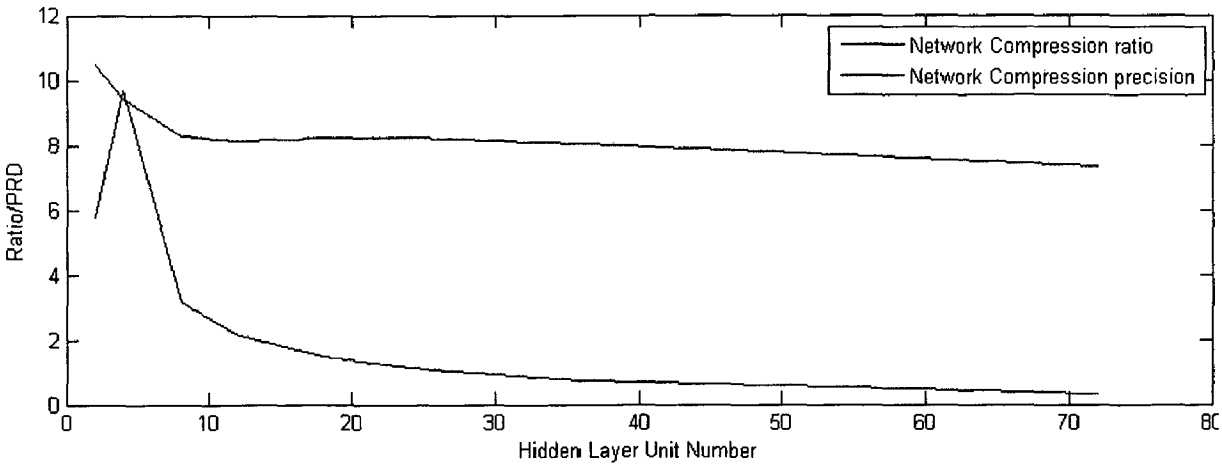



**Figure6.4 Effect of hidden layer size on PRD and comp. ratio / record cmba217**

## 6.4 Comparison of the proposed technique with existing techniques

Before comparing the performance of the proposed ECG compression technique with the existing ones, let us see, first, the basis of comparison.

According to Jalaleddine *et al*[11], some of the factors which should be considered for ensuring a solid basis of comparison among ECG compression techniques are:

a) Identical application lead bandwidth(i.e. monitoring, diagnostics, or Holter),

b) All data should be from standard databases (i.e. like MIT-BIH, AHA etc).

c) The reconstructed ECG signals must meet or exceed specific error criteria for ECG segments and waves. These error criteria must be clinically application dependent.

One more important point that is highlighted by Jalaleddine *et al*[11] is, the employment of the PRD in evaluating ECG compression schemes has no practical value. Although the rms error between original waveforms and reconstructed waveforms is a common form of comparison, it does not reveal whether or not an algorithm can preserve diagnostically significant features of the ECG waveform.

With these basic understandings I will present the comparison of the proposed technique, i.e. ECG data compression using ANN, with the existing ECG data compression techniques. The Network performances of the proposed technique are presented. The Overall performances are left because the aim of the dissertation is to study if ANN can be used for ECG data compression, I will focus my discussion on the Network performances (i.e. Network compression ratio and Network compression precision).

The comparison of the proposed compression system and existing techniques discussed in chapter 4 is summarized in table 6.7.

**Table 6.7 comparison of proposed techniques with existing ones**

| Method | Compression ratio | signal frequency (Hz) | signal precision (bits) | PRD (%) | comments |
|---|---|---|---|---|---|
| AZTEC | 10.0 | 500 | 12 | 28.0 | Poor P and T Fidelity |
| TP | 2.0 | 200 | 12 | 5.3 | Sensitive to signal frequency(SF) |
| CORTES | 4.8 | 200 | 12 | 7.0 | Sensitive (SF) Poor P Fidelity |
| Fan/SAPA | 3.0 | 250 | -- | 4.0 | High Fidelity |
| Entropy Coding of $2^{nd}$-difference ECGs | 2.8 | 250 | 10 | -- | susceptible to transmission errors |
| peak-picking (spline) with entropy coding | 10.0 | 500 | 8 | 14.0 | Limited Results |
| DPCM (delta coding with threshold) | 4.0 | 300 | 8 | -- | Sensitive (SF) Poor P Fidelity |
| DPCM Linear prediction | 2.5 | 250 | 12 | -- | High Fidelity |
| DPCM-Linear Prediction Interpolation, and Entropy coding | 7.8 | 500 | 8 | 3.5 | Sensitive (SF) and quantization |
| Orthogonal Transforms (CT, KLT,HT) | 3.0 | 250 | -- | -- | Lead-I |
| Dual application of K-L Transformation | 12.0 | 250 | 12 | -- | (X,Y,Z) Leads |
| Fourier Descriptors | 7.4 | 250 | 12 | 7.0 | (X,Y) Leads |
| **The Proposed Technique** | **10.0** | **360** | **11** | **8.30** | **High fidelity** |

According to the reported Network compression ratios from the 33 experiments I have carried out, the average Network compression ratio is 10:1. This value is better or equal to all the reported compression ratios of the existing ECG data compression techniques, but Dual application of K-L Transformation (see table 6.7). However one more advantage of the proposed ECG technique is that, the compression ration increases significantly as the length of the data increases (i.e. as more beats are used) (see table 6.2).

The average Network compression precision (in terms of PRD), according to the reported PRDs of all the 33 experiments, is 8.30. This value implies, the proposed compression technique is more precise than AZTEC and peak-picking (spline) with entropy coding. However it also seems to imply that it is a little less precise than CORTES and Fourier Descriptors and significantly less precise than TP, Fan/SAPA, and DPCM-Linear Prediction Interpolation, and Entropy coding.

But According to Jalaleddine *et al*[11], the CORTES and Fan/SAPA are tested using idealized ECG waveforms, rather than standard databases. Hence it will be difficult to compare the compression precisions reported using these techniques with the proposed technique. Moreover, it was reported in [11] that the CORTES deteriorated substantially whenever it was used with sampling rates higher than 200Hz while the proposed technique is less frequency dependent.

Even though the TP algorithm and DPCM-Linear Prediction Interpolation, and Entropy coding look better in compression precision, it is reported that they are sensitive to signal frequency and quantization. In addition the TP algorithm gives poor P-wave fidelity.

In the proposed technique it is reported that the ECG wave components are reconstructed with good fidelity. Hence ECG data compression using ANN is a better and promising technique in the future if more researches are done in this field.

# CHAPTER SEVEN

# CONCLUSION AND RECOMMENDATION

## 7.1 Conclusion

As the results in Chapter 6 show, Backpropagation artificial neural network based compression system is an efficient way of ECG data compression. It gives an improved compression ratio and compression precision. The compression of ECG data using this technique is real-time.

The existing ECG data compression techniques such as AZTEC, CORTES, FAN AND SAPA all give a compression ration less than 10. But using this technique it is common to achieve a compression ration greater than 20. The interesting thing about the proposed compression technique is, as the size of the data increases, more similar ECG signals will be represented by few weights and hidden values. Therefore for long time ECGs like ambulatory ECG this technique is very successful.

If we compare the compression precision of the proposed technique and the existing ECG data compression techniques, this technique gives a satisfactory precision of 6.69. This value is the average of all cycles in a single record plus all the 33 experiments. This calculation of the compression precision may lower the actual compression power of the system. So it is better to compare this parameter using same data for all techniques.

One of the crucial components of ECG used in ECG interpretation is the R-R interval. The proposed compression system is also investigated for R-R interval error. As table 6.1 shows

this error is significantly low. Hence we can conclude that the proposed compression system does not affect ECG interpretations directly or indirectly related to this ECG component.

One of the good features of using ANN based compression system is, there is only a little reduction or addition in the number of sample points in an ECG data. In other words most of the sample points of the original ECG are used to train a BP ANN; which is the main reason why this technique yields high fidelity. However many of the ECG compression techniques discussed in Chapter 2 alter the sample points by a significant amount. This will considerably affect the signal retrieval.

A draw back of using Backpropagation artificial neural network based compression system could be the distortion of the peak values (R –points) of the ECG data. The main reason might be the low number of sample points around the QRS complex of ECGs. As a result enough amounts of points may not be available to exactly predict the peak. But since the peak values (R-points) are very important in ECG interpretation, this problem should be solved. The remedy used in this work is to save the original ECG peaks and replace the reconstructed peaks by the original one. This will require an R-point detection of the reconstructed data. This solution may decrease the compression ratio and may increase the compression time by few amounts; but it is tolerable.

Even though it is not compared with the existing ECG compression techniques, one of the successes of the proposed technique is the representation of the onsets and offsets of the P-waves and T-waves of the original ECG data in the reconstructed ECG data. The results in table 6.2 show these fiducial points are well represented in the retrieved signals.

## 7.2 Recommendation

BP algorithm usually reaches convergence after hundreds of iterations. If the amount of work in each training is cut down, then it will benefit shortening the compression time greatly.

If we combine Backpropagation (BP) algorithm with the existing compression techniques discussed in Chapter 4, such as turning point (TP) algorithm, it would be possible to cut down the compression time. TP algorithm has high speed of execution and high fidelity. Its compression ratio is about 2:1. If we use TP algorithm to process data before using BP algorithm, the number of points in each ECG cycles after using TP algorithm is half the number before. The input and the output neural unit number are half the before, which cut down the amount of work of each training and shorten the compression time greatly.

This combination of TP algorithm with BP algorithm not only decreases the processing time but also the compression ration. Because if the neural input and output numbers are reduced, the network size will be reduced which means the size of the weights to be stored will effectively be reduced.

Moreover, the performance of the artificial neural network can be improved if techniques like pre-grouping are used before compression. This technique is will be helpful to group ECG complexes into sets of signals which are morphologically similar to one another. The more similar the set of ECG beats to be compressed by the ANN are the better the results be, in terms of compression performance (high compression ratio, low reconstruction error).

Self-organizing ANN models such as, competitive learning networks, Fuzzy ART and Fuzzy Min-Max cluster are some of the options which can be used to perform pre-grouping.

# References


1. John G Webster, *Medical Instrumentation Application and Design*, John Wiley & sons (Asia) Pre Ltd., 1999
2. Leslie Cromwell, Fred J. Weibell, Erich A. Pfeiffler, *Biomedical Instrumentation and Measurements*, Prentice Hall of India, 1993.
3. Tracy Barill, The *Six Second ECG: A Practical Guidebook to ECG Interpretation*, Nursecom Educational Technologies (USA), 2004
4. URL: http://www.cvphysiology.com/Arrhythmias/A013.htm, posted by Richard E. Klabunde, Ph.D. Last visited June 1, 2008
5. URL: http://www.doc.ic.ac.uk/~nd/surprise_96/journal/vol4/cs11/report.html#Contents , posted by Christos Stergiou and Dimitrios Siganos, Last visited May 29, 2008.
6. Hagan, M.T., H.B. Demuth, and M.H. Beale, *Neural Network Design*, Boston, MA: PWS Publishing, 1996.
7. Hagan, M.T., H.B. Demuth, and M.H. Beale, *Matlab Neural Network Toolbox™ 6 User's Guide*, The MathWorks, Inc.,2008
8. Richard.P.Lippmann, "An Introduction to Computing with Neural Networks " IEEE ASSP MAGAZINE APRIL 1987.
9. Simon Haykin, *Neural Networks, a comprehensive foundation*, Second Edition, Prentice Hall, New Jersey, 1999.
10. S.N Sivanandavm, S. Sumathi, S.N. Deepa, *Introduction to Neural Networks using MATLAB 6.0*, Tata McGraw Hill, India , 2006
11. Sateh M S Jaldeddine, Chriswell G Hutchens, Robert D Strattan, and William A Coberly, "ECG data compression techniques - a unified approach", *IEEE Transactions on Biomedical Engineering,* vol. 37, no. 4, pp. 329-343, April 1990.
12. C. A. Andrews, J. M. Davies, and G. R. Schwarz, "Adaptive data compression," *Proc. IEEE,* vol. *55,* pp. 267-277, Mar. 1967.
13. M. Shridhar and N. Mohankrishnan, "Data compression techniques for electrocardiograms", *Can. Elec. Eng. J.,* vol. 9, no. 4, pp. 126- 131, 1984.
14. Special issue on redundancy reduction, *Proc. IEEE,* vol. 55, Mar. 1967
15. data compression," in *Proc. IRE Nut. Symp. Space Electron. Telemetry,* IEEE LG-SET REC., Sect. 4.1, 1962, pp. 1-10 ],
16. J. E. Medlin, "Sampled- data prediction for telemetry bandwidth compression," *IEEE Trans. Space Electron. Telem.,* vol. SET-I 1, pp. 29-36, Mar. 1965.

17. G. Benelli, V. Cappellini, and F. Lotti, "Data compression techniques and applications," *Radio Electron. Eng.,* vol. 50, no. 1 /2, pp. 29-53, 1980.
18. U. E. Ruthann and H. V. Pipberger, "Compression of the ECG by prediction or interpolation and entropy encoding," *IEEE Trans. Biomed. Eng.,* vol. BME-26, pp. 613-623, Nov. 1979.
19. C. M. Kortman, "Redundancy reduction-a practical method of data compression," *Proc. IEEE,* vol. 55, pp. 253-263, Mar. 1967.
20. P. Elias, "Predictive coding-Part I and Part 11," *IRE Trans. Inform.Theory,* vol. IT-1, pp. 16-33, Mar. 1955.
21. C. E. Shannon, "A mathematical theory of communication," *Bell* Sysr. *Tech. J.* , vol. 27, pp. 379-423. July 1948.
22. J . R. Cox, F. M. Nolle, H. A. Fozzard, and G. C. Oliver, "AZTEC, a preprocessing program for real-time ECG rhythm analysis," *IEEE Trans. Biomed. Eng.,* vol. BME-15, pp. 128-129, Apr. 1968.
23. J . R. Cox, H. A. Fozzard, F. M. Nolle, andG. C. Oliver, "Some data transformations useful in electrocardiography," in *Computers and Biomedical Research vol. 111,* R. W. Stacy and B. D. Waxman, Eds. vol. 111. New York: Academic, 1974, pp. 181-206.].
24. J . R. Cox, F. M. Nolle, and R. M. Arthur, "Digital analysis of the electroencephalogram, the blood pressure wave, and the ECG," *Proc. IEEE,* vol. 60, pp. 1137-1164, Oct. 1972.
25. W. C. Mueller, "Arrhythmia detection program for an ambulatory ECG monitor," *Biomed. Sci. Instrument.,* vol. 14, pp. 81-85, 1978.
26. J . P. Abenstein and W. J . Tompkins, "New data-reduction algorithm for real-time ECG analysis," *IEEE Trans. Biomed. Eng.,*vol. BME-29, pp. 43-48, Jan. 1982.
27. N. J. Holter, "New method for heart studies," *Science,* vol. 134, pp. 1214-1220, 1961.
28. N. Ahmed and K. R. Rao, *Orthogonal Transforms for Digital Signal Processing.* New York: Springer, 1975.
29. K. R. Rao and N. Ahmed, "Orthogonal transforms for digital signal processing," *IEEE Internat. Con$* ASSP, pp. 136-140, 1976.
30. D. F. Elliott and K. R. Rao, *Fast Transforms: Algorithms, Analysis and Applications.* New York: Academic, 1982.
31. Ji Zhenyan, Deng Shanxi , "A Compression system of ECG Data Based on Neural Network, *Proceedings of ICSP*, 1998.
32. Friesen GM, Jannett TC, Jadallah MA, et al. *A Comparison of the Noise Sensitivity of Nine QRS Detection Algorithms.* IEEE Transactions on Biomedical Engineering. Vol. 37. No.1 January 1990. p85-98.

33. Riedmiller, M., and Braun H., "A direct adaptive method for faster backpropagation learning: The RPROP algorithm," *Proceedings of the IEEE International Conference on Neural Networks*, 1993.
34. URL: MIT-BIH Arrhythmia Database at Physionet.org: http:// physionet.org /physiobank/ database /mitdb/, last updated Updated Monday, 05-Nov-2007 13:05:34 EST
35. URL: http://en.wikipedia.org/wiki/Ecg#Limb, last modified on 2 June 2008, at 05:18
36. THE "ALAN E. LINDSAY ECG TUTORIAL" V6.0 (July 2007) Frank G. Yanowitz, MDProfessor of Medicine University of Utah School of Medicine Director, IHC ECG Services LDS Hospital Salt Lake City, Utah.
37. Zipes, Libby, Bonow, Braunwald, BRAUNWALD'S HEART DISEASE, a text book of cardiovascular medicine, 7th ed., Elsevier Inc. 2005.
38. URL: http://www.merck.com/mmhe/sec03/ch027/ch027a.html, Last full review/revision January 2008 by L. Brent Mitchell, MD
39. URL: http://www.coldbacon.com/mdtruth/path.html, last visited on June 16 2008.