

Master Thesis

Waqar Ahmed

Deep Learning of Cardiac Related Condition using a Non-Contact Multi-Sensor System

Chair for Medical Information Technology

Faculty of Electrical Engineering and Information Technology, RWTH Aachen University

Univ.-Prof. Dr.-Ing. Dr. med. Steffen Leonhardt

Supervisor: Dr.-Ing. Anake Pomprapa, Dr.-Ing. André Stollenwerk

Date: 10. Januar 2018

Acknowledgments

Eidesstattliche Versicherung

Name, Vorname

Matrikelnummer (freiwillige Angabe)

Ich versichere hiermit an Eides Statt, dass ich die vorliegende Arbeit/Bachelorarbeit/
Masterarbeit* mit dem Titel

selbstständig und ohne unzulässige fremde Hilfe erbracht habe. Ich habe keine anderen als die angegebenen Quellen und Hilfsmittel benutzt. Für den Fall, dass die Arbeit zusätzlich auf einem Datenträger eingereicht wird, erkläre ich, dass die schriftliche und die elektronische Form vollständig übereinstimmen. Die Arbeit hat in gleicher oder ähnlicher Form noch keiner Prüfungsbehörde vorgelegen.

Ort, Datum

Unterschrift

*Nichtzutreffendes bitte streichen

Belehrung:

§ 156 StGB: Falsche Versicherung an Eides Statt

Wer vor einer zur Abnahme einer Versicherung an Eides Statt zuständigen Behörde eine solche Versicherung falsch abgibt oder unter Berufung auf eine solche Versicherung falsch aussagt, wird mit Freiheitsstrafe bis zu drei Jahren oder mit Geldstrafe bestraft.

§ 161 StGB: Fahrlässiger Falscheid; fahrlässige falsche Versicherung an Eides Statt

(1) Wenn eine der in den §§ 154 bis 156 bezeichneten Handlungen aus Fahrlässigkeit begangen worden ist, so tritt Freiheitsstrafe bis zu einem Jahr oder Geldstrafe ein.

(2) Straflosigkeit tritt ein, wenn der Täter die falsche Angabe rechtzeitig berichtigt. Die Vorschriften des § 158 Abs. 2 und 3 gelten entsprechend.

Die vorstehende Belehrung habe ich zur Kenntnis genommen:

Ort, Datum

Unterschrift

Abstract

Traditional sensors which required a direct contact with the body are old fashioned now, as they make trouble in the daily activities. A non-contact multi-sensors has been implemented to show the power of non contact sensors.

pilot usecase has been considered while making the system.

A CNN model has also trained to detect various cardiac conditions in a real-time environment. A cloud based interface has been set up so the the health status can be tracked from any where in the world.

Table of Contents

Acknowledgments	iii
Abstract	vii
Table of Contents	ix
List of Figures	xiii
List of Tables	xv
Symbolverzeichnis	xvii
1 Abbreviations	xvii
1 Introduction	1
1.1 Motivation	1
1.2 Literature Review	2
1.3 Aim	3
1.4 Objective	4
2 Background	7
2.1 Anatomy of Heart	7
2.2 The Electrocardiogram	8
2.3 ECG Complex	11
2.3.1 P Wave	11
2.3.2 QRS Complex	11
2.3.3 T Wave	12
2.3.4 PR Interval	12
2.3.5 ST Segment	12
2.3.6 QT Interval	13
2.4 Disadvantages of Attached Electrodes	13
2.5 Noise in ECG Signal	13
2.5.1 Power Line Interference	13
2.5.2 Baseline wander	14
2.5.3 Muscle Noise	14
2.6 Arrhythmias	15
2.6.1 Causes of Arrhythmia	15
2.6.2 Types of Arrhythmias	15
3 Signal Processing and Feature Extraction	19
3.1 Devices	19
3.1.1 Magnetic Induction Sensor	19
3.1.2 Photoplethysmogram Sensor	22
3.1.3 ECG Sensor	22

Table of Contents

3.1.4	Ballistocardiogram Sensor	23
3.1.5	Thermal Camera	23
3.2	ECG Signal Processing	23
3.2.1	Filter Method	24
3.2.2	Artificial Intelligence Method	24
3.2.3	Wavelet Transform Method	24
3.3	Wavelet Transform	24
3.4	Mallat's Algorithm	25
3.5	Filter Bank Construction	26
3.6	Dataset	32
3.7	Preprocessing	33
3.7.1	Wavelet Transform Method	33
3.7.2	Band-pass Filter Method	34
3.8	QRS Detection	35
3.9	P and T Wave Detection	40
3.10	Heart Rate Calculation	41
3.11	Algorithm Execution on ECG Chair Data	44
3.12	PPG Signal Processing	45
4	Deep Learning	47
4.1	Machine Learning	47
4.1.1	Supervised Learning	47
4.1.2	Unsupervised Learning	47
4.2	Artificial Neural Networks	49
4.2.1	Artificial Neuron	49
4.2.2	Activation Function	50
4.3	Convolutional Neural Network	51
4.3.1	Convolutional Layer	51
4.3.2	Pooling Layer	52
4.3.3	Fully Connected Layer	52
4.4	Keras	53
4.5	Convolutional Neural Network for the Identification of Cardiac Arrhythmia	55
4.5.1	Layers Explanation	55
4.5.2	Results	57
5	Visualization and System Architecture	59
5.1	Grafana	59
5.2	InfluxDB	60
5.3	Setting up Grafana with InfluxDB	60
5.4	Lambda Architecture for Big Data Processing	62
5.4.1	Batch Layer	63
5.4.2	Speed Layer	63
5.4.3	Serving Layer	64
5.5	System Architecture	66
5.5.1	Data Layer	67

5.5.2	Speed Layer	68
5.5.3	Serving Layer	68
5.6	Device Management Interface	70
6	Conclusion	73
6.1	Future Work	74
References		75

List of Figures

2.1	The electrical activity of heart[Sch17].	8
2.2	The leads position on chest[Gre].	9
2.3	The axillary lines on the right side of chest[Com17a].	10
2.4	The possible position of limb leads[Com17b].	10
2.5	The ECG signal[Com17c].	11
2.6	AC Interference of 50Hz[mau17].	14
2.7	Baseline wandering in ECG signal[mau17].	14
2.8	ECG signal combined with muscle noise[mau17].	15
2.9	Premature Ventricular Contraction[con17b].	16
2.10	Atrial Fibrillation, Atrial Flutter and Tachycardia[sum17].	17
2.11	LBBB vs RBBB[bil17].	18
3.1	A set of sensor devices.	20
3.2	Two-channel filter bank	26
3.3	The ECG signals from MIT-BIH dataset.	32
3.4	The filtered ECG signal using wavelet.	34
3.5	The filtered ECG signal using bandpass filter.	34
3.6	ECG signal and its decomposition at different scales.	35
3.7	Maximum and minimum values on scale 3.	36
3.8	Maximum and minimum values pair for finding the R peaks.	37
3.9	Detected R peaks.	37
3.10	Flowchart for locating R peaks.	38
3.11	Flowchart for locating Q and S peaks.	39
3.12	Detected Q, R and S peaks.	40
3.13	Detected P,Q,R,S and T waves.	41
3.14	Flowchart for locating P wave.	42
3.15	Flowchart for locating T wave.	43
3.16	Original ECG signal from the chair.	44
3.17	Detected P,Q,R,S and T waves on the chair's ECG signal.	44
3.18	The denoised PPG signal using band-pass filter.	45
4.1	Basic supervised machine learning workflow.	48
4.2	An artificial neural network with its 3 basic layers [con17a].	49
4.3	A single artificial neuron.	50
4.4	Activation functions [ujj17].	51
4.5	Convolutional neural network example.	52
4.6	Max pooling layer example.	53
4.7	A 6-layer Convolutional Neural Network model for the identification of cardiac arrhythmia.	54
4.8	Layers used in the model and number of parameters to be optimized.	55
4.9	A set of four subfigures.	56

5.1	InfluxDB data source setup in Grafana.	61
5.2	Dashboard setup.	61
5.3	Interface to add query for the panel.	62
5.4	Lambda architecture.	63
5.5	Using more data from serving layer in case of speed layer having a problem.	64
5.6	Lambda symbol representing the lambda architecture.	65
5.7	Implementation of lambda architecture.	65
5.8	System architecture for the sensors.	66
5.9	Using the deep learning model in real-time environment.	67
5.10	Real-time ECG signal visualization.	68
5.11	Real-time MI signal visualization.	69
5.12	Real-time PPG signal visualization.	69
5.13	Real-time ECG signal visualization on android application.	70
5.14	The device management interface.	71

List of Tables

3.1	Attributes of MI sensor.	21
3.2	Byte identifiers for the MI sensor.	21
3.3	Attributes of PPG sensor.	22
3.4	Wavelet transform ECG signal frequency range[SLL14].	35
4.1	Training data.	57
4.2	Test data.	58
4.3	False prediction counts.	58

Symbols

1 Abbreviations

AV	Atrioventricular
ANN	Artificial Neural Network
CNN	Convolutional Neural Network
DWT	Discrete Wavelet Transform
ECG	Electrocardiogram
FFT	Fast Fourier Transfrom
HRC	Heart Rate Variability
LBBB	Left Bundle Branch Block
PVC	Premature Ventricular Contraction
RBBB	Right Bundle Branch Block
RMSE	Root Mean Square Error
RT	Repolarization Time
SA	Sinoatrial
WT	Wavelet Transform

1 Introduction

Around one billion people travel on airlines annually [ML08], [F⁺03]. It has also been predicted that the number will be doubled in two decades. During a flight, emergencies occur at a rate of 20 to 100 per million passengers. Many of the cases are not even on the record as there is no proper reporting system. The most common in-flight complaints relate to respiratory, cardiac, traumatic or gastrointestinal related cases. Out of these, cardiac and respiration related complaints are the most serious. During in-flight medical emergencies, a doctor is present only 30 to 60 percent of the time [ML08], [Dow00]. This number may have changed as the article was published in 2008.

1.1 Motivation

Environmental changes such as the rising of altitude, the level of oxygen gradually decreases as the air weakens, including the reduction of atmospheric pressure, temperature, and humidity. As a result, the heart rate increases as it tries to deliver more oxygen to the muscles, which can lead to fainting or even heart attacks among some passengers.

It is important to realize that the on-board medical resources are limited. Therefore, a technological advancement is required, which can reduce the workload of doctors during a flight. Healthcare is one of the hottest research areas in this era. Monitoring of vital signs, parameter, like respiration, ECG, EEG, temperature, and heart rate are of great importance.

A variety of technologies are already available for measuring the vital signs [LU18]. They include traditional stethoscopes, electrodes for measuring ECG, and different types of gauges, but they have their drawbacks in terms of comfort and convenience. For example, to measure the ECG, the electrodes are required to be directly attached to the skin of a patient, which is very inconvenient and limits the patient's movement. Blood pressure measurement using Sphygmomanometer that uses belts or cuffs to measure the blood pressure, which again limits the daily activities. Even though these technologies are reliable and provide better results, but they are inadequate for long-term everyday activities.

Contactless sensors are the next big topic in the healthcare. Multiple vital signs can be measured with these sensors without any need of direct contact with the body. They are designed in such a way that they can be integrated into the daily surroundings without any disruption. Different techniques have been used to integrate contact-less sensors into bathtubs [LKP04], chairs [AL07], smartwatches, smartphones, toilet seats [KLP04], and beds [WZ08] to measure different vital signs.

The distributed computing, streaming analytics, and machine learning have become more

powerful, cheaper, and faster [McD18], and they can be applied in various industries:

- Healthcare
- Transportation
- Automobile
- Manufacturing
- Retail

The combination of streaming data, big data analysis, and machine learning can benefit healthcare for identifying chronic diseases such as cardiac disease. Vital signs of the patient can also be analyzed in real-time. The integration of non-contact sensor, and deep learning technologies can be used to identify the cardiac arrhythmias in a real-time environment during the flight.

1.2 Literature Review

Variety of methods and devices are available to measure vital signs. The majority of these contributions based on direct contact with the skin. Jeong et al. [JYK05] measures the blood pressure. They measure the pulse wave using a PPG sensor, which was attached to the earlobe, and an ECG monitoring device with electrodes. Usually electrodes are used to measure the signals from the human body [SSLO07], [Neu98].

Many attempts have been made to use sensors that do not require direct contact with the body, but still, they depend on dry electrodes which do not require gel. Jin-Chern Chiou et al. in their study [CKL⁺06] showed that how they used the fabricated dry electrodes to measure the EEG signal. Their results showed that dry electrodes perform comparably to the conventional electrodes, but the problem with this approach is that they are limited to only specific areas of the body with no hair where the contact is good.

In the last few years, non-contact sensors have gained popularity and have been conspired to measure the signals. Thomas et al. [SDC07] presented a gel-free, non-contact ECG/EEG sensor that capacitively coupled to the skin and can operate up to 3mm distance to the skin. Professor S. Leonhardt [AL07] described a technique to measure the ECG signal using capacitive coupled electrodes, integrated into an office chair. The signal was measured through a shirt without any direct contact with the skin. Kin-fai Wu et al. [WZ08] in their work proposed a heart rate monitoring system based on a bed, which used contactless electrodes to measure the ECG signal. The design is based on a bedsheets, which is made

up of highly conductive material, together with a separate measuring device, which can measure the ECG signal of a lying subject through clothes.

Electronics company muRata have created under-the-bed sensor [?]. The sensor uses BCG principle and uses an accelerometer to capture the micro movements caused by respiration and heart. The sensor can measure heart rate, respiration rate, heart rate variability and stroke volume.

Yong Kyu Lim et al. [LKP04] measured the ECG signal using insulated electrodes. The electrodes were attached to the bathtub on both sides of the chest. The recorded signal in their study was noisier as compared to the conventional electrodes signal. But the R peaks were large enough to be detected, which can help to get various vital signs. Yong Kyu Lim et al. in their another study [KLP04] measured the ECG signal on a toilet seat. The capacitive coupled electrodes were used that was insulated on a toilet seat.

Many researchers have used previously traditional machine learning techniques to classify the ECG signals, but the problem with that approach is that the model depends on the researcher's understanding of the data, which is a huge burden. Recent advancement in deep learning techniques has attracted the researchers to implement these techniques in the healthcare.

In 2016 Jun et al. [JPM⁺16] proposed a deep neural network to recognize premature ventricular contraction (PVC) beats in an ECG signal. A deep neural network with 6 hidden layers was trained using the TensorFlow library to classify PVC and normal ECG signals. Pourbabae et al. [PRK16] trained a deep convolutional neural network to classify the normal ECG signals with paroxysmal atrial fibrillation (PAF).

In the above both studies, the deep neural network can only classify two different ECG signals. Isin et al. [IO17] used a transferred deep convolutional neural network to classify three different types of ECG signals. In their study, they have used a deep learning framework AlexNet (Krizhevsky et al., 2012) that was previously trained on the general image dataset to carry out the classification of ECG signals.

1.3 Aim

The aim of this thesis is to build a software system that provides a healthcare solution, which would be able to track various vital signs such as heart rate, temperature, and ECG. Moreover, it should be able to detect the cardiac arrhythmias in real-time with a deep learning model.

The non-contact multi-sensors system consists of the following sensors:

1. Capacitive ECG sensor
2. Photoplethysmogram sensor
3. Magnetic impedance sensor
4. Ballistocardiogram sensor
5. Thermal camera

There can be many use cases where this system can be implemented such as trains, buses, and cars. The use case which is focused in this thesis is the aircraft where the vital parameters of the pilot can be measured.

This system is integrated with non-contact multi-sensors in order to monitor the vital parameters and cardiac related conditions of a pilot during the course of a flight. The early identification of the disease can help to provide a proper treatment to the pilot, as well as can stop from reaching any dangerous situation.

An arrhythmia can be harmless or life threatening. Therefore, for a pilot, a thorough medical evaluation is necessary to assess the severity of arrhythmia for the safety of both pilots and the passengers.

A deep learning model has been designed based on a convolutional neural network in order to detect the cardiac conditions in real-time. The model can detect 4 different types of ECG signals. Various cardiac arrhythmia datasets have taken from the existing dataset. The advantage of using CNN is that, unlike other machine learning algorithms, it does not require a feature extraction phase.

Multiple platforms have been used along with tablets to visualize the results in hand in real-time. A cloud has been set up to store all the sensors data and vital parameters so the data can be accessed from anywhere in the world. The sensor's data can also be used for different purposes such as for the re-training of the deep learning model.

1.4 Objective

The objectives of this thesis are as follow:

- Programming of a software visualization (cECG, MI, BKG, and PPG) for a PC along with the tablet notification and visualization
- Construction of the data bank on the cloud

- Preprocessing of the signals and feature extraction
- Arrhythmia data collection
- Deep learning of the cardiac conditions
- Evaluation of the algorithm with real-time data from the non-contact multi-sensors system

2 Background

2.1 Anatomy of Heart

The function of the heart is to pump the blood inside the body, which is stimulated by an electrical stimulus [Wil05] [Sch17]. The heart pumps blood to the different parts of the body such as organs, muscles, and tissues.

The heart shown in Figure 2.1 is made up of 4 chambers, left and right atria, and left and right ventricles. The right atrium receives de-oxygenated blood from the whole body and pumps it into the right ventricle which then pumps the blood to the lungs for increasing oxygenation. The left atrium receives oxygenated blood from the lungs and pumps it into the left ventricle which then pumps the oxygenated blood to the whole body. The aorta carries oxygenated blood to the different part of the body and the pulmonary arteries carry the de-oxygenated blood back to the lungs for improving oxygenation. The blood flows to different organs via arteries and returns back to the heart via veins.

The main components of the cardiac conduction system are:

1. Sinoatrial (SA) node
2. The Atrioventricular (AV) node
3. Atrioventricular (AV) bundle or bundle of His
4. Right and left bundle branches
5. Purkinje fibers

The SA node, also known as sinus node, is a natural pacemaker of the heart which is located in the right atrium. It produces an electrical stimulus at the rate of 60 to 100 signals per minute (under normal condition), which travels through the heart to make it pump the blood to the body. It initiates all heartbeats and determines the heart rate. The electrical impulse from the SA node spreads throughout the atrium which results in the contraction of the atrium. The AV node which is located on the other side of the right atrium, near the AV valve, serve as a gateway to the ventricles. It also delays the passage of electrical impulse to the ventricles. This is to ensure that all the blood is ejected from the atria to the ventricles before the ventricles contract. The AV node then passes the signal to the atrioventricular (AV) bundle or bundle of His. The bundle is divided into two parts, right and left bundle branches to stimulate the right and left ventricles. The signal then travels down to the Purkinje fibers, where the signal spreads upwards throughout the ventricular myocardium. Each contraction of the ventricles represents one single heartbeat.

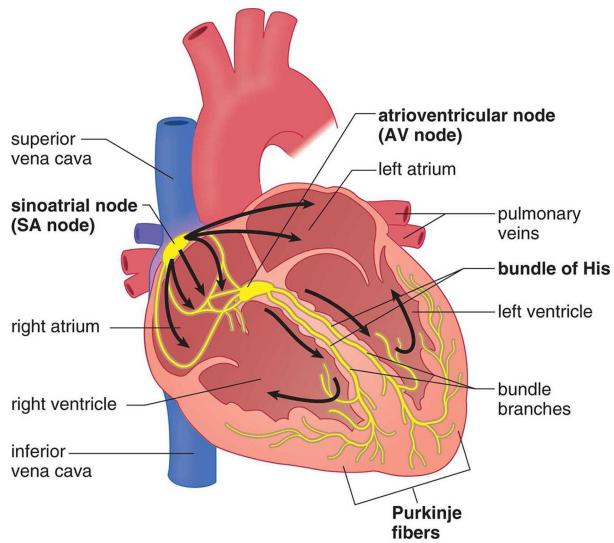


Figure. 2.1: The electrical activity of heart [Sch17].

Each heartbeat is composed of two phases, known as systole and diastole. During **systole**, the heart muscles contract and the blood is pumped from ventricles to the different parts of the body. During **diastole**, the hearts muscles relax and the blood from atria flows into the ventricles. The pressure generated during systole from the ventricular contraction is high, whereas, during diastole, due to the muscle relaxation this pressure reduces.

The electrical activity of the heart can be detected in the form of electrocardiogram by placing electrodes on the body surface. It is a powerful tool for diagnosing the status of patient's heart.

2.2 The Electrocardiogram

Electrocardiogram (ECG) is an essential tool for diagnosing the electrical activity of the heart [Wil05]. It is a simple and non-invasive procedure to measure the activity of the heart. Most of the tools available today to measure the ECG are based on the electrodes which are required to be attached to the body. The electrodes sense the electrical potentials of the body and transmit them to the ECG monitor. These voltages are then transformed into appropriate waveforms which represent the heart's polarization and depolarization cycle. Different components of the wave represent the activity of different parts of the heart.

In conventional 12-lead ECG, only 10 electrodes or leads are attached to the patient's body and the electrical activity of the heart can be viewed from 12 different perspectives [cab17]. These 12 views of the heart are captured by placing the electrodes on chest, wrists, and ankles. The main purpose of ECG is to identify any cardiac arrhythmia, ischemia, problems

with heart rate or any other irregularities.

These 10 electrodes are divided into 2 groups.

1. 6 chest electrodes
2. 4 limb electrodes

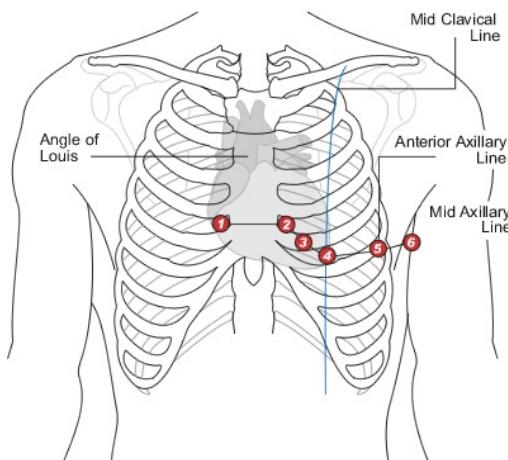


Figure. 2.2: The leads position on chest[Gre].

Chest Electrodes

The chest electrodes are denoted as “V” and they all are numbered from V1 to V6 as can be seen in Figure 2.2. The electrodes are positioned at the following locations on the chest:

- V1 - Fourth intercostal space (between ribs 4 and 5) on the right sternum
- V2 - Fourth intercostal space (between ribs 4 and 5) on the left sternum
- V3 - Placed in the middle of V2 and V4
- V4 - Fifth intercostal space (between ribs 5 and 6) at the mid-clavicular line
- V5 - Placed horizontally on anterior axillary line with V4
- V6 - Placed horizontally on Mid-axillary line with V4 and V5

The 3 different axillary lines **anterior axillary line**, **midaxillary line**, and **posterior axillary line** can be seen in Figure 2.3.

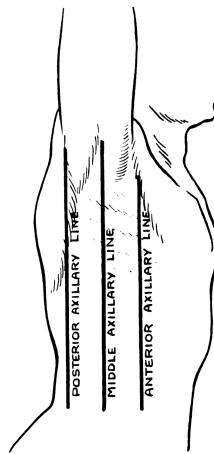


Figure. 2.3: The axillary lines on the right side of chest[Com17a].

Limb Electrodes

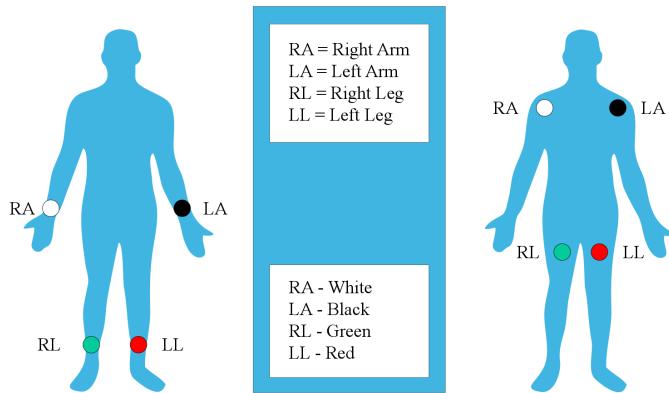


Figure. 2.4: The possible position of limb leads[Com17b].

The 4 limb electrodes are denoted as RA, LA, LL, RL and their respective positions are:

- RA - Anywhere on right arm between shoulder and elbow, but avoiding thick muscles.
- LA - Symmetric to the RA position, but on left arm
- RL - Anywhere on right leg between the torso and the ankle
- LL - Symmetric to the RL position, but on left leg

The limb electrodes are shown in Figure 2.4

2.3 ECG Complex

ECG complex represents the electrical activity of the heart during one cardiac cycle [Wil05]. A normal cardiac cycle consists of five waveforms labeled with P, Q, R, S and T as can be seen in Figure 2.5. The Q, R and S waves are referred to as one unit, the QRS complex. The ECG signal represents the conduction of electrical impulses from the atria to the ventricles. The important parameters in the ECG signal are:

2.3.1 P Wave

The P wave is the first component of the ECG signal. It occurs due to contraction of both left and right atrium. This process is also known as atrial depolarization. A normal P wave has following characteristics (in lead II):

- Duration: less than 120 milliseconds
- Amplitude: less 0.25 mV in the limb leads and less than 0.15 mV in the precordial leads
- Location: before the QRS complex

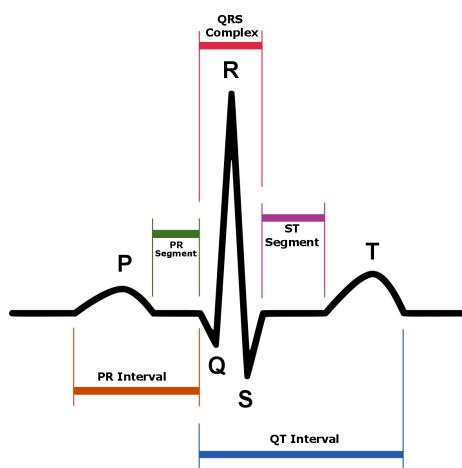


Figure. 2.5: The ECG signal[Com17c].

2.3.2 QRS Complex

The QRS complex follows the P wave and represents a contraction of both right and left ventricles. This contraction results in the blood ejection from the heart which eventually pumps into the arteries and creates a pulse. The Q and S waves are relatively very

2 Background

small whereas, R wave is comparatively very big. A normal QRS complex has following characteristics (in lead II):

- Duration: 70 to 100 milliseconds
- Location: follows the P wave

2.3.3 T Wave

The T wave represents the ventricles repolarization. It occurs during the last part of ventricle systole. The T wave has following characteristics (in lead II):

- Duration: 100 to 250 milliseconds or greater
- Location: follows the QRS complex

2.3.4 PR Interval

The PR interval is the time interval between the end of contraction of the atria and the beginning of contraction of the ventricles. A normal PR interval has following characteristics:

- Duration: 120 to 200 milliseconds
- Location: From the beginning of P wave to the beginning of the QRS complex

2.3.5 ST Segment

The ST segment represents the end of ventricular depolarization and the beginning of the ventricles relaxation. The point between the end of QRS complex and the beginning of ST segment is called as the J point. A normal ST segment has following characteristics:

- Duration: 80 to 120 milliseconds
- Location: From the end of QRS complex to the beginning of T wave

2.3.6 QT Interval

The QT interval is the time interval between the ventricular depolarization and repolarization. The QT duration varies according to the heart rate. Faster heart rate results in smaller QT interval whereas, slower heart rate may result in a bigger QT interval. The bigger QT interval may result in an irregular heartbeat. A normal QT interval has following characteristics:

- Duration: 360 to 440 milliseconds
- Location: From the beginning of QRS complex to the end of T wave

2.4 Disadvantages of Attached Electrodes

While it is easy to monitor the electrical heart activity by placing the electrodes directly on the body but it has several disadvantages as well.

1. It limits the patient's mobility.
2. Discomfort for the patient as electrodes are directly attached to the body.
3. Loss of cardiac monitoring in case if patient moves.
4. Long time contact of the electrodes may irritate the skin.

2.5 Noise in ECG Signal

Typically the ECG signal can be corrupted by the different types of noises and artifacts which changes the characteristics of the ECG signal [LD]. Hence, it is difficult to extract the useful information from the signal. Following are the major noises, which are present in the ECG signal.

2.5.1 Power Line Interference

Power line interference is a 50 Hz noise, which is present in ECG signal because of the improper grounding of the ECG equipment or interference from the nearby equipment. In order to remove this type of noise, a proper use of a filter is required. A 50 Hz notch filter

can be used to remove the power line interference. Figure 2.6 illustrates the 50 Hz power line interference in ECG signal.

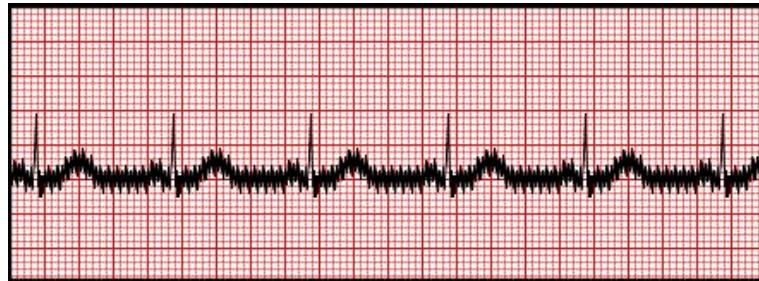


Figure. 2.6: AC Interference of 50Hz[mau17].

2.5.2 Baseline wander

Baseline wander is a low-frequency component which corrupts the ECG signal because of breathing, body movements, dirty or lose electrodes or electrode impedance. Generally, they have a frequency lower than 2 Hz [Sch12]. A high pass filter can be used to remove the baseline wander. The baseline wandering in ECG signal can be seen in Figure 2.7.



Figure. 2.7: Baseline wandering in ECG signal[mau17].

2.5.3 Muscle Noise

This type of noise is caused by muscle contractions besides heart, which results in the change of heart electrical potential [MG13]. Whenever the other muscles near the electrodes depolarized and repolarized, they also generate waves, which can be monitored by the ECG. They generally occur in short time burst and have higher amplitude values than the ECG signal. It can be removed using Wavelet transform [MPS⁺11]. An example of ECG signal affected by muscle contractions can be seen in Figure 2.8.



Figure. 2.8: ECG signal combined with muscle noise[mau17].

2.6 Arrhythmias

Irregularity in the heartbeat is known as arrhythmia (also called dysrhythmia) [med17]. During an arrhythmia, a heart is out of normal rhythm and may feel like the heart has skipped a beat or beat with an irregular pattern. A normal heart rate lies between 60 to 100 beats per minute and arrhythmia can occur with normal heart rate, slow heart rate (called bradycardia), in which heart rate is less than 60 beats per minute or with rapid heart rate (called tachycardia) in which heart beats faster than 100 beats per minute.

2.6.1 Causes of Arrhythmia

Arrhythmia can be caused by one of the following reasons:

- Heart disease
- Electrolyte imbalance
- Changes in heart muscle
- After surgery effects

2.6.2 Types of Arrhythmias

The most common types of arrhythmias are:

Premature Ventricular Contraction

It is one type of arrhythmia, in which the heartbeat is initiated by the ventricles rather than the SA node. It is generally referred as “skipped beat”. In PVC, there is no P wave prior to the PVC beat, and the ST-T wave appears in the opposite direction to the QRS complex. Moreover, the RR interval between the preceding R peak and PVC is shorter than the normal. The sinus beat after the PVC occurs on schedule. This is the most common type of arrhythmia, which occurs with or without any heart disease. It could be the result of too much stress, usage of too much cocaine or restless. Sometimes it can also be caused by heart disease. However, most of the time PVC is considered as harmless and rarely needs a treatment.

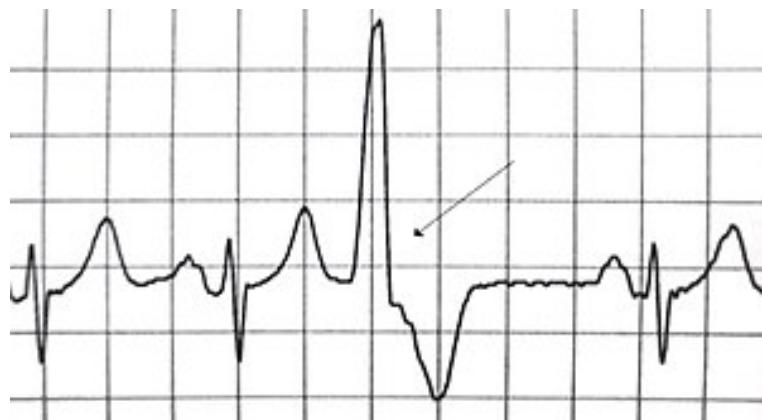


Figure. 2.9: Premature Ventricular Contraction[con17b].

Atrial Fibrillation

This type of arrhythmia caused by the abnormal contraction of the upper chamber of the heart. During atrial fibrillation, the atria beat irregularly without any coordination with the ventricles. This could results in heart palpitation, shortness of breath and weakness.

Atrial Flutter

This type of arrhythmia caused by problems in the heart's electrical system. It is similar to atrial fibrillation but rhythm in atria is more organized than the atrial fibrillation. The risk factors and causes of atrial flutter are similar to those of atrial fibrillation.

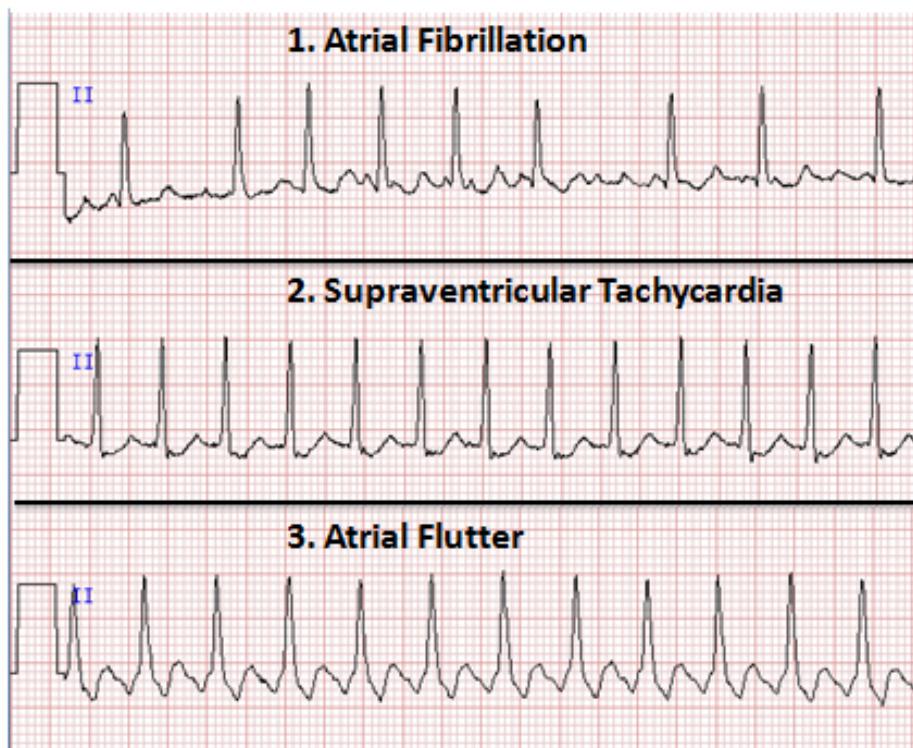


Figure. 2.10: Atrial Fibrillation, Atrial Flutter and Tachycardia[sum17].

Bradycardia

In this type of arrhythmia, the heart beats slower than the normal pace, usually less than 60 beats per minute. This could be because of the disease in electrical heart system.

Tachycardia

In this type of arrhythmia, the heart beats faster than the normal pace, usually, more than 100 beats per minute. When the heart beats too fast, it may not pump blood effectively to the body parts, which could result in shortness of breath.

Heart Block

In this type of arrhythmia, the heart beats slowly because of the delay or complete block of the electrical signal between the upper chambers and the lower chambers of the heart. It is also called atrial ventricular block (AV block).

Bundle Branch Block

Bundle branch block can be of two types, left bundle branch block (LBBB) and right bundle branch block (RBBB). In a normal heart, both bundles depolarized simultaneously and contract at the same time. In this type of arrhythmia, the affected bundle depolarized slowly whereas the un-affected bundle depolarized normally which results in a broader QRS complex, generally longer than 120 milliseconds duration. LBBB and RBBB can be seen in Figure 2.11. In RBBB, the delayed activation of the right ventricle gives rise to the ST depression and the T wave inversion in the lead V1 and the wide S wave in the lead V6. Similarly, in LBBB, the delayed activation of the left ventricle, results in the absence of Q wave and tall R wave appeared as M shaped in lead V6 and deep S wave in the lead V1 [cab17].

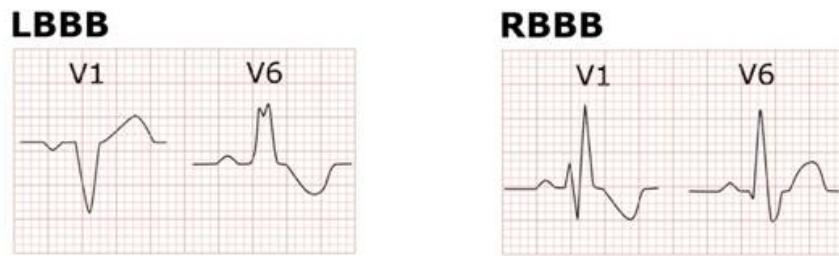


Figure. 2.11: LBBB vs RBBB[bil17].

3 Signal Processing and Feature Extraction

Wireless sensor devices and contact-less sensor devices are the current trends in the health informatics. The recent improvements in the sensor devices made it possible for the people to bring this idea into reality. When medical sensor devices are combined with cloud computing, it can be thought of as a complete solution for a healthcare system which can be used not only in hospitals but also can be utilized out of the hospital. These sensors can monitor some vital signs such as body temperature, respiration, heart rate, blood pressure, ECG or EEG.

3.1 Devices

Multiple non-contact sensors are used to implement a system for this thesis, which collects data of the user and processes them in real-time. The following devices are used in this work:

3.1.1 Magnetic Induction Sensor

The magnetic induction (MI) sensor uses the magnetic fields to measures the small changes in electrical resistance of the thorax. A magnetic coil is used which induces eddy currents within the thorax. This eddy current reinduces a secondary magnetic field which can be measured with the same or another coil. The amplitude of the eddy current is directly proportional to the magnetic flux density and the conduction of the material. Based on this concept, the thorax conductivity, which changes according to the inflation and deflation of the lungs, enable us to capture the breathing of the patient [MRF13].

The MI sensor provides a data packet of 42 bytes, which splits into the attributes shown in Table 3.1 and the byte identifier is shown in Table 3.2.

As mentioned in Table 3.2, 0x81 and 0x82 represents the byte identifier, but in some cases, it also presents in the data as they are simply the hexadecimal number. The sensor algorithm replaces the data value 0x81 by 0x8101 and 0x82 by 0x8102 in order to differentiate the data values from the byte identifiers. Therefore, to process sensor data correctly, the hexadecimal value 0x8101 should be replaced by 0x81 and 0x8102 by 0x82.

3 Signal Processing and Feature Extraction

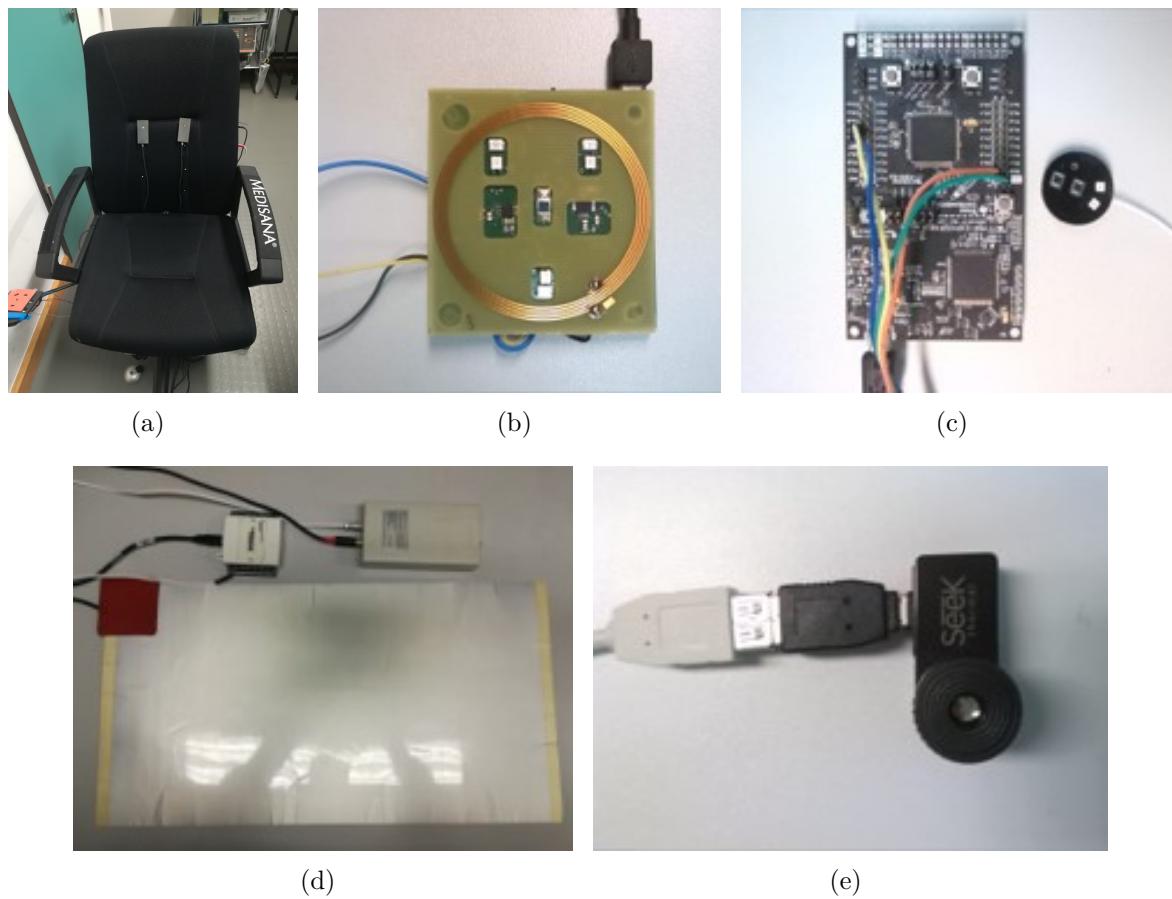


Figure. 3.1: A set of sensor devices: (a) an ECG sensor; (b) a MI sensor; (c) a PPG sensor; (d) a BCG sensor; and, (e) a thermal camera.

Tab. 3.1: Attributes of MI sensor.

Attributes	Size (Bytes)
MI_RAW	4
MI	4
RED_RAW	4
ECG_RAW	4
IR_RAW	4
RED_AVG	4
ECG_AVG	4
IR_AVG	4
ACC_X	2
ACC_Z	2
ECG_REF	2
RESP_REF	2
BATTERY	2

Tab. 3.2: Byte identifiers for the MI sensor.

2-Byte Identifier	Description
0x81	Header
0x82	Data Packet

3.1.2 Photoplethysmogram Sensor

The photoplethysmogram (PPG) sensor is used to measures the variations in blood flow in the body with each heartbeat. It is an optical, simple, low cost and non-invasive technique, which can measure various vital signs at the surface of the skin. A PPG sensor uses a light source to illuminate the skin and a photo-detector to measures the variations in the light intensity associated with changes in the oxygen level in blood. The decrease in light intensity indicates the increase in oxygen level in blood, whereas, increase in light intensity indicates the decrease of oxygen level in blood. They can provide valuable information about the cardiovascular system such as the pulse rate.

The sensor provides PPG signals with 4 different channels, a temperature, and accelerometer coordinates. The size of the data changes according to the attributes, as shown in Table 3.3. The temperature value is sent every one second, while, the frequency rate of PPG channels is 100 samples per second. For the accelerometer coordinates, the data rate is 50 samples per second.

Tab. 3.3: Attributes of PPG sensor.

2-Byte Identifier	Attributes	Size (Bytes)	Data
0x0050	ppg (8 Bytes)	2	Channel 1
		2	Channel 2
		2	Channel 3
		2	Channel 4
0x0054	Temperature (2 Bytes)	2	Temperature
0x0041	Accelerometer coordinates (6 Bytes)	2	X Coordinate
		2	Y Coordinate
		2	Z Coordinate

3.1.3 ECG Sensor

As described in Section 2.2, the ECG signal is usually collected by placing electrodes directly on the body but it has several disadvantages as described in Section 2.4. Therefore,

non-contact capacitive electrodes have been used to collect the ECG signals. Unlike traditional electrodes, which rely on galvanic contact, the capacitive electrodes are insulated from skin using a dielectric material, such as air gap or clothes [Bou17]. The ECG signal propagates via skin to the dielectric material and then to the electrodes through a capacitive coupling. The major drawback of this approach is that it is very sensitive to a body motion.

3.1.4 Ballistocardiogram Sensor

The ballistocardiogram (BCG) sensor measures the ballistic forces associated with cardiac contraction and ejection of blood. These ballistic forces are mainly measured by the electromechanical film (EMFi), which converts the mechanical energy into the electrical signal. Most of the time, the EMFi sensing device is placed on a chair or bed, which measures the pressure associated with the cardiac activity.

3.1.5 Thermal Camera

A thermal camera is also used to measure the temperature of the person. A thermal Seek camera captures the thermo temperature images from which the temperature is calculated.

3.2 ECG Signal Processing

The detection of QRS complex is the basis for processing ECG signal. Regardless of what application is required, the accurate detection of QRS complex is a prerequisite for feature extraction. In order to detect the QRS complex accurately, it is necessary to detect the R-peak position correctly. Once the QRS complex is identified, further operations can also be performed on the signal such as heart rate calculation, classification of ECG signal and P and T detection.

The “QRS Complex” is the combination of Q, R and S waves and it represents the contraction of the ventricles. It plays a significant role in the detection of cardiac arrhythmias.

Many methods have already been proposed for the detection of QRS complex. These methods fall into 3 categories [PZZ10]:

3.2.1 Filter Method

The filter method uses a bandpass filter to filter the ECG signal [PT85][RSN97]. In this method, a QRS complex is intensified by suppressing the P and T wave. This method is generally very quick and takes less time to implement. But the major drawback of this method is that the frequency band of QRS complex and of noise overlapping affect its performance.

3.2.2 Artificial Intelligence Method

The detection of QRS complex using this method is fast, accurate and more robust, but in reality, it is time-consuming and difficult to implement [XHT92][Pie91][CSCB90]. Therefore, this method is not very popular and not widely used as compared to the other methods.

3.2.3 Wavelet Transform Method

Wavelet transform method becomes popular in detecting the QRS complex. It is based on time-frequency analysis. It is efficient and takes less to implement. Many researchers have already used wavelet transform for detecting the QRS complex. Yazhu Qiu [QDFM06] used Mexican-hat wavelet to detect ECG signal. In the proposed method, although the processing was fast, it sometimes did not detect the onset and offset of QRS complex accurately. Nevertheless, it is considered as faster and easier to implement and it is used to extract the ECG signal features in this thesis.

3.3 Wavelet Transform

A wavelet transform is a very useful tool for analyzing the signal simultaneously in both time and frequency domain [Add17]. It uses a little wavelike functions known as **wavelets**. Wavelets are used to transform a signal into another representation where signal information can be viewed in a more useful form.

Generally, there are two operations involved with wavelet. Either they can be stretched or squeezed or can be translated to other locations on the signal and if the wavelet matches the shape of the signal at specific location or scale, it produces a large transform value. And similarly, if the signal and the wavelet do not correlate, it produces a low transformed value. There is a single function called “mother wavelet” which is stretched or translated

to produce a family of basis functions known as “daughter wavelet”. A mother wavelet is defined as:

$$\Psi_{a,b}(t) = \frac{1}{\sqrt{|a|}} \Psi\left(\frac{t-b}{a}\right), \quad a, b \in \mathbb{R}, a \neq 0 \quad (3.1)$$

where a is the scaling parameter which measures the degree of the scale, and b is the translation parameter, which measures the time location of the wavelet. If $|a| < 1$, then it mainly corresponds to higher frequencies. And on the other hand, if $|a| > 1$, it corresponds to lower frequencies. It is important to note here that the variation in time and frequency scale of the wavelet is supervised by the Heisenberg's uncertainty principle. At large scale, the time domain is not very clear, whereas, in the frequency domain is much finer. As the scale decreases, the frequency domain becomes worse, whereas, time domain becomes finer.

The wavelet transform of a signal $f(x)$ is defined as:

$$X_W(a, b) = \frac{1}{\sqrt{|a|}} \int_{-\infty}^{\infty} \left(\frac{t-b}{a} \right) f(x) dx \quad (3.2)$$

If $a = 2^j (j \in Z)$, then the wavelet transform is called the dyadic wavelet transform or the binary wavelet transform [MH92]. To calculate the dyadic wavelet transform of a signal, a mother wavelet is first chosen and then dilated by a power of two. The Quadratic Spline wavelet is used with ECG signal as the mother wavelet. It has a property that the sharp edges occur at the zero crossing point in the transformation, which corresponds to the local maxima or minima of the smoothed signal. The dyadic wavelet transform of a signal $f(n)$ can be calculated by using Mallat's Algorithm [PZZ10].

3.4 Mallat's Algorithm

Mallat algorithm [MH92] for the signal $f(n)$ is defined as follows:

$$s_{2^j} f(n) = \sum h_k s_{2^{j-1}} f(n - 2^{j-1}k), \quad (3.3)$$

$$w_{2^j} f(n) = \sum g_k s_{2^{j-1}} f(n - 2^{j-1}k). \quad (3.4)$$

where, $s_{2^0}f(n)$ is the original signal to be processed, in this case, it is the ECG signal. $w_{2^j}f(n)$ is the wavelet transform of the input signal $f(n)$ at scale 2^j . h_k and g_k are the coefficients of a low-pass filter and high-pass filter respectively. The wavelet transform of a signal $f(n)$ is obtained by passing the input signal through a digital filter bank, which consists of a low-pass and high-pass filters. The design and implementation of the Wavelet transform is based on perfect reconstruction filter banks. The coefficients of low-pass filter $H(z)$ and high-pass filter $G(z)$ can be calculated by constructing a filter bank.

3.5 Filter Bank Construction

A biorthogonal filter bank is designed based on Quadratic Spline wavelet. A filter bank is a set of filters and sampling operators [SN96]. The downsampling operators are decimators, whereas, upsampling operators are expanders. In a 2-channel filter bank, the analysis and synthesis phase both contain a low-pass and a high-pass filter. These filters are $H_0(z)$ and $H_1(z)$ for the analysis phase, and $G_0(z)$ and $G_1(z)$ for the synthesis phase respectively [WLH01], as shown in Figure 3.2. After passing the input signal $X[n]$ from the filters, the resulting signal is first down-sampled by 2 and then up-sampled by 2 respectively, producing the final output signal $Y[n]$.

The filters $H_0(z)$ and $H_1(z)$ are not ideal filters, therefore, their responses overlap, there is aliasing in each channel and there is distortion. The synthesis filters $G_0(z)$ and $G_1(z)$ should be adjusted according to the analysis filters $H_0(z)$ and $H_1(z)$ to cancel out the error.

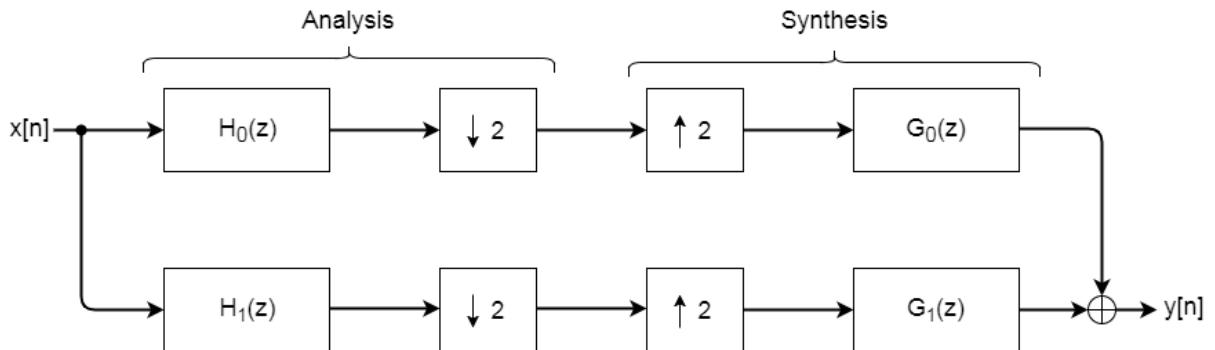


Figure. 3.2: Two-channel filter bank

The idea is to determine H_0 , H_1 , G_0 and G_1 such that, $Y[n]$ is a delayed version of input signal $X[n]$. This is called as perfect reconstruction filter bank. A perfect reconstruction filter bank is also known as “biorthogonal” and the associated filters as biorthogonal filters. A perfect reconstruction with an l -step delayed in z domain without upsampling and downsampling represented as:

$$H_0(z)G_0(z) + H_1(z)G_1(z) = z^{-l} \quad (3.5)$$

The combination of downampler ($\downarrow 2$) and an upsampler ($\uparrow 2$) removes the odd number components. In z domain, it can be written as:

$$(\downarrow 2)(\uparrow 2)H_0X = \frac{1}{2}[H_0(z)X(z) + H_0(-z)X(-z)] \quad (3.6)$$

It contains only the even components as the odd components have already been removed. The term $H_0(-z)X(-z)$ is an aliasing term. This aliasing term has to cancel the aliasing effect from the other channel to remove the alias.

After passing the input signal from the channel 1, it produces:

$$Y_0(z) = \frac{1}{2}G_0(z)[H_0(z)X(z) + H_0(-z)X(-z)] \quad (3.7)$$

Similarly, for the 2nd channel, it produces:

$$Y_1(z) = \frac{1}{2}G_1(z)[H_1(z)X(z) + H_1(-z)X(-z)] \quad (3.8)$$

Adding the output of these 2 channels produce the final output.

$$\begin{aligned} Y(z) &= Y_0(z) + Y_1(z) \\ &= \frac{1}{2}G_0(z)[H_0(z)X(z) + H_0(-z)X(-z)] + \frac{1}{2}G_1(z)[H_1(z)X(z) + H_1(-z)X(-z)] \end{aligned} \quad (3.9)$$

Arranging $Y(z)$ in such a way so that, one part depends on $X(z)$ and the other part on $X(-z)$. We get,

$$Y(z) = \frac{1}{2}[G_0(z)H_0(z) + G_1(z)H_1(z)]X(z) + \frac{1}{2}[G_0(z)H_0(-z) + G_1(z)H_1(-z)]X(-z) \quad (3.10)$$

In Equation 3.10, the part that depends on $X(-z)$ is the aliasing part, and the part that depends on $X(z)$ is the distortion part.

3 Signal Processing and Feature Extraction

The perfect reconstruction for filter bank can be achieved if the following two conditions are satisfied.

1. No aliasing:

$$G_0(z)H_0(-z) + G_1(z)H_1(-z) = 0 \quad (3.11)$$

2. No distortion:

$$G_0(z)H_0(z) + G_1(z)H_1(z) = mz^{-k} \quad (3.12)$$

where m is a constant and k is a time delay.

In order to satisfy condition 1 i.e., to get rid of aliasing, choose:

$$\begin{aligned} G_0(z) &= H_1(-z), \\ G_1(z) &= -H_0(-z) \end{aligned} \quad (3.13)$$

Substituting the values of Equation 3.13 in 3.11 yields:

$$H_0(-z)H_1(-z) - H_0(-z)H_1(-z) = 0 \quad (3.14)$$

which implies to 0, hence the aliasing is removed.

In order to remove the distortion, i.e, to satisfy equation 3.12, lets assume that,

$$P_0(z) = G_0(z)H_0(z) \quad (3.15)$$

This is a low-pass product filter. The high-pass product filer is $P_1(z) = G_1(z)H_1(z)$. The relationship between the terms P_0 and P_1 can be deduce by Equation 3.13:

$$P_1(z) = G_1(z)H_1(z) = -H_0(-z)G_0(-z) = -P_0(-z) \quad (3.16)$$

The equation 3.12 can be re-written as:

$$P_0(z) - P_0(-z) = mz^{-k} \quad (3.17)$$

The design of 2-channel filter bank is now reduced to only two steps:

1. Design a low-pass filter which satisfy Equation 3.17.

2. Factorize P_0 to find the values of H_0 and G_0 .

Equation 3.17 can be further simplified by normalizing $P_0(z)$ by z^k to center it. Substituting $P(z)$ by $z^k P_0(z)$ and $P(-z)$ by $-z^k P_0(-z)$ produces:

$$P(z) + P(-z) = m \quad (3.18)$$

Equation 3.18 implies that $P(z)$ is a half-band filter, in which all even powers of z in $P(z)$ are zero except for the constant term (coefficient of the term z^0). Furthermore, all the odd powers cancel out when $P(z)$ and $P(-z)$ are added.

One possibility to design the low-pass filter $P_0(z)$ is to use the Daubechies construction [SN96]:

$$P_0(z) = (1 + z^{-1})^{2p} Q(z) \quad (3.19)$$

$P_0(z)$ is called “maxflat filter” where p can be any integer and has $2p$ coefficients, i.e., the length of the filter. $(1 + z^{-1})^{2p}$ is called *binomial filter* and it is a spline filter, and $Q(z)$ be a polynomial of degree $(2p - 2)$ is chosen such that $P_0(z)$ satisfy the half-band filter property. Moreover, the order of $P_0(z)$ is always an even number. p defines the number of zeros to be placed at π in a unit circle.

Any value of p can be chosen and the polynomial $Q(z)$ depends on the value of p , i.e., $Q(z)$ should be chosen in such a way that it satisfies the half-band filter property. If $p = 1$, then this represents the Haar case and it has 2 coefficients.

In this case, $p = 2$ is chosen. Therefore, the number of coefficients are 4 and degree of $Q(z)$ be 2. Then Equation 3.19 becomes:

$$P_0(z) = (1 + z^{-1})^4 Q(z) \quad (3.20)$$

Let $Q(z)$ be of form,

$$Q(z) = a + bz^{-1} + az^{-2} \quad (3.21)$$

Substituting Equation 3.21 in 3.20 produces:

$$P_0(z) = a + (4a + b)z^{-1} + (7a + 4b)z^{-2} + (8a + 6b)z^{-3} + (7a + 4b)z^{-4} + (4a + b)z^{-5} + az^{-6} \quad (3.22)$$

To satisfy the condition on $P_0(z)$, equating all the coefficients of odd power of z to 0, except for the center term, i.e., term at z^{-3} which equates to 1, produces:

$$4a + b = 0 \quad (3.23)$$

$$8a + 6b = 1 \quad (3.24)$$

Solving Equation 3.23 and 3.24 yields:

$$a = \frac{-1}{16}, b = \frac{1}{4} \quad (3.25)$$

From Equation 3.20, 3.21 and 3.25, we get:

$$P_0(z) = \frac{(1 + z^{-1})^4(-1 + 4z^{-1} - z^{-2})}{16} \quad (3.26)$$

Factorizing $P_0(z)$ to get $H_0(z)$ and $G_0(z)$:

$$\begin{aligned} H_0(z) &= \frac{(1 + z^{-1})^3}{4} \\ &= \frac{(1 + 3z^{-1} + 3z^{-2} + z^{-3})}{4} \end{aligned} \quad (3.27)$$

$$\begin{aligned} G_0(z) &= \frac{(1 + z^{-1})(-1 + 4z^{-1} - z^{-2})}{4} \\ &= \frac{(-1 + 3z^{-1} + 3z^{-2} - z^{-3})}{4} \end{aligned} \quad (3.28)$$

Then by equation 3.13, we have:

$$\begin{aligned} H_1(z) &= G_0(-z) \\ &= \frac{(-1 - 3z^{-1} + 3z^{-2} + z^{-3})}{4} \end{aligned} \tag{3.29}$$

$$\begin{aligned} G_1(z) &= -H_0(-z) \\ &= \frac{(-1 + 3z^{-1} - 3z^{-2} + z^{-3})}{4} \end{aligned} \tag{3.30}$$

Therefore, the filter coefficients are:

$$\begin{array}{llll} h_0(0) = & \frac{1}{4} & h_0(1) = & \frac{3}{4} \\ h_0(2) = & \frac{3}{4} & h_0(3) = & \frac{1}{4} \\ h_1(0) = & \frac{-1}{4} & h_1(1) = & \frac{-3}{4} \\ h_1(2) = & \frac{3}{4} & h_1(3) = & \frac{1}{4} \\ g_0(0) = & \frac{-1}{4} & g_0(1) = & \frac{3}{4} \\ g_0(2) = & \frac{3}{4} & g_0(3) = & \frac{-1}{4} \\ g_1(0) = & \frac{-1}{4} & g_1(1) = & \frac{3}{4} \\ g_1(2) = & \frac{-3}{4} & g_1(3) = & \frac{1}{4} \end{array} \tag{3.31}$$

The coefficients of filters are just simple fractions, therefore, the amount of computation and time needed for the transformation is small. As our interest is only in the decomposition of the signal, therefore, we will consider only the H_0 and H_1 filters coefficients.

Let $f(n)$ be the ECG signal, then the wavelet transform of $f(n)$ can be calculated by using Mallat algorithm as follows:

$$s_{2^0}f(n) = f(n) \tag{3.32}$$

$$s_{2^j}f(n) = \frac{1}{4}s_{2^{j-1}}f(n) + \frac{3}{4}s_{2^{j-1}}f(n - 2^{j-1}) + \frac{3}{4}s_{2^{j-1}}f(n - 2^j) + \frac{1}{4}s_{2^{j-1}}f(n - 2^j * 3) \tag{3.33}$$

$$w_{2^j}f(n) = \frac{-1}{4}s_{2^{j-1}}f(n) + \frac{-3}{4}s_{2^{j-1}}f(n - 2^{j-1}) + \frac{3}{4}s_{2^{j-1}}f(n - 2^j) + \frac{1}{4}s_{2^{j-1}}f(n - 2^j * 3) \quad (3.34)$$

where $w_{2^j}f(n)$ is the Biorthogonal Spline Wavelet Transform of ECG signal at scale 2^j .

3.6 Dataset

The MIT-BIH Arrhythmia dataset is used for the implementation of the system. It contains 48 hours of recording of 47 subjects. Each record contains 2 signals, namely MLII and V5, with a recording of 30 minutes duration. The sample rate for the recording is 360 samples per second per channel with 11-bit resolution over a 10mV range. Each record consists of 3 files:

- Header file (.hea): It contains information such as the number of samples, sampling frequency, ECG signal format, number of ECG leads and their types, patient's history and the detailed clinical information.
- ECG signals (.dat): It contains the original signal values of both MLII and V5 leads. The signals from MLII lead are used in the analysis.
- Attribute file (.atr): It contains the annotation information of the ECG signal, annotated by the doctors.

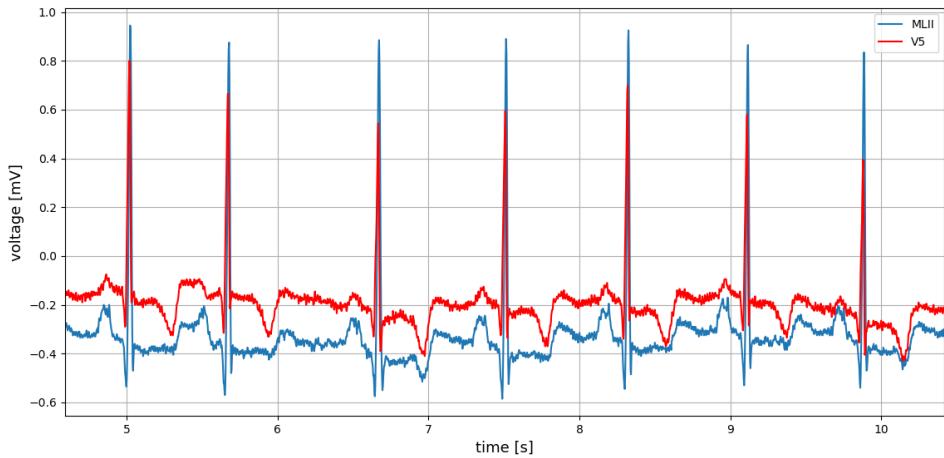


Figure. 3.3: The ECG signals from MIT-BIH dataset.

There is a specific python package *wfdb-python* available for reading the data from the MIT-BIH dataset. The ECG signals of one of the patients can be seen in Figure 3.3. It contains 2 signals, namely, MLII and V5.

The signals are generally contaminated with noise or baseline drift. Therefore, they are required to be processed in order to achieve better signals.

3.7 Preprocessing

Two different methods have been used to remove the noise and artifacts from the signal in the system implementation.

1. Wavelet transform method
2. Band-pass filter method

3.7.1 Wavelet Transform Method

The wavelet transform decomposes the signal into different frequency bands by passing the signal through high-pass and low-pass filter respectively, which results in 2 sets of coefficients namely, approximation coefficients and detail coefficients. The approximation coefficients contain the low-pass filter coefficients and the detail coefficients contain the high-pass filter coefficients. The next step is to split the approximation coefficients again into 2 parts using the same procedure and so on.

The original signal contains the high-frequency noise and the baseline drift. The wavelet transform can be used to remove the corresponding noises and the baseline drift. The process starts by decomposing the original signal into 8 layers using Wavelet type *bior2.6*, which results in the corresponding detail and approximation coefficients. Mostly, the layers 1 and 2 of the detail coefficients contain the high-frequency noise and the layer 8 of the approximation coefficients contain the baseline drift. Therefore, the layers 1 and 2 of the detail coefficients and layer 8 of the approximation coefficients are set to 0, which then results in the de-noised signal with no baseline drift. The resulting ECG signal is shown in Figure 3.4.

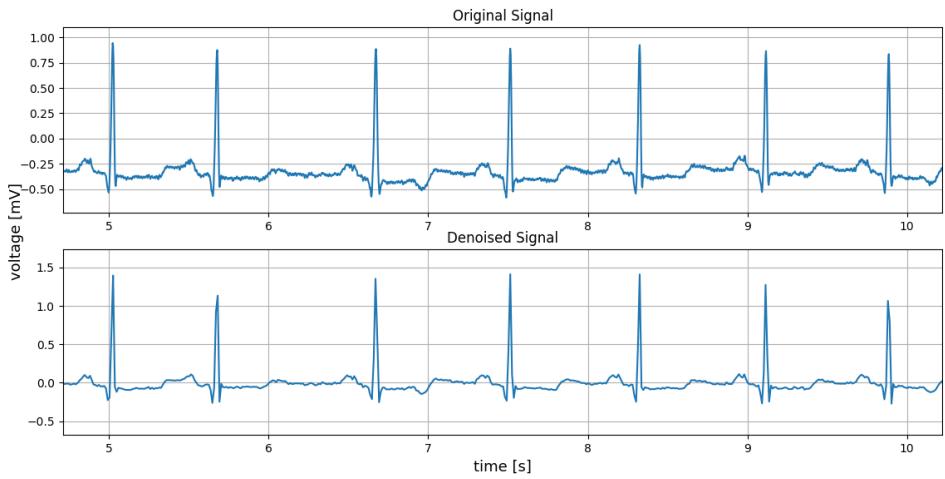


Figure. 3.4: The filtered ECG signal using wavelet.

3.7.2 Band-pass Filter Method

A band-pass filter is a type of filter, which passes frequencies only in a certain range or spread without disturbing the input signal. A band-pass filter can be used to reduce the baseline drift, motion artifacts and high-frequency noise from the ECG signal. A passband of 0.5 Hz to 40 Hz has been used. After applying the band-pass filter, this results in the denoised signal with no high-frequency and baseline drift. The denoised signal is shown in Figure 3.5.

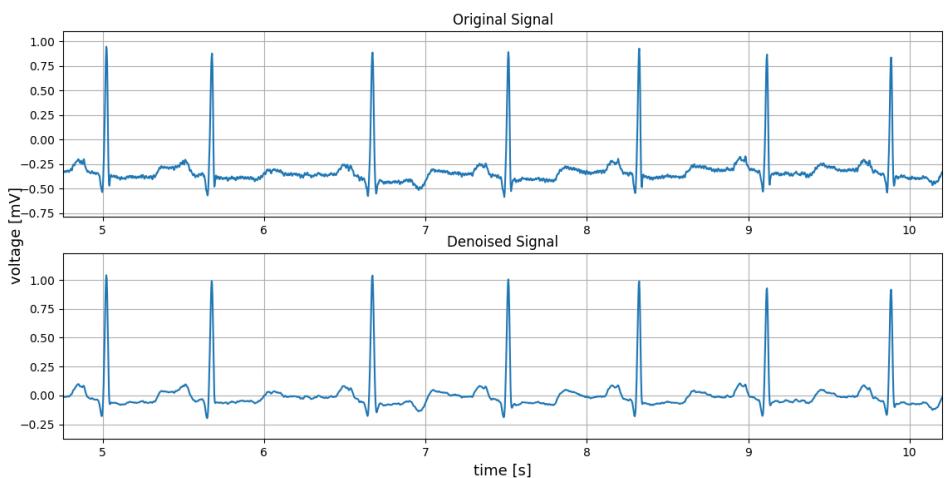


Figure. 3.5: The filtered ECG signal using bandpass filter.

3.8 QRS Detection

From Equation 3.33 and 3.34, the Biorthogonal Spline Wavelet Transform of ECG signal is obtained, which is the first derivative of the signal at different scales. The signal is decomposed into 4 levels, as shown in Figure 3.6. The procedure starts by first locating the R peaks and then the onset and offset of QRS complexes, and finally the onset and offset of P and T waves. Due to Wavelet characteristic derivative, this wavelet transforms the signal maxima and minima into the zero crossing point at the different scales, which makes it easier to detect the location of P, QRS and T waves [E10].

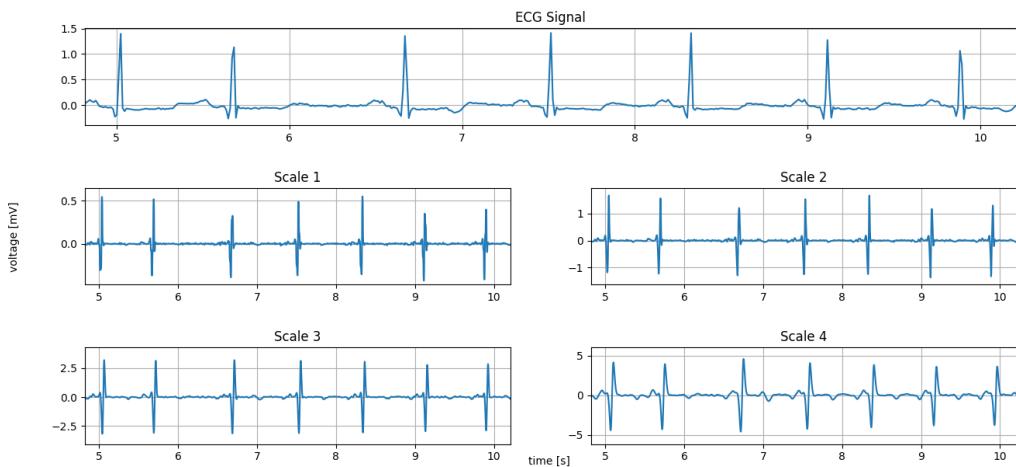


Figure. 3.6: ECG signal and its decomposition at different scales.

Tab. 3.4: Wavelet transform ECG signal frequency range[SLL14].

Transform Scale	Frequency Range (Hz)
2^1	90.0~180.0
2^2	29.92~84.24
2^3	1.52~38.88
2^4	5.76~19.44

Most of the QRS complex energy lies between 3 Hz and 40 Hz, and it corresponds to the 3rd scale as shown in Table 3.4. The R peaks correspond to the zero-crossing point of the modulus maxima pair, with a little delay, therefore, first all the positive and negative extrema on scale 3 be identified. The extrema are shown in Figure 3.7. To reduce the influence of noise and other interference in locating the R peaks, the scale 3 signal is

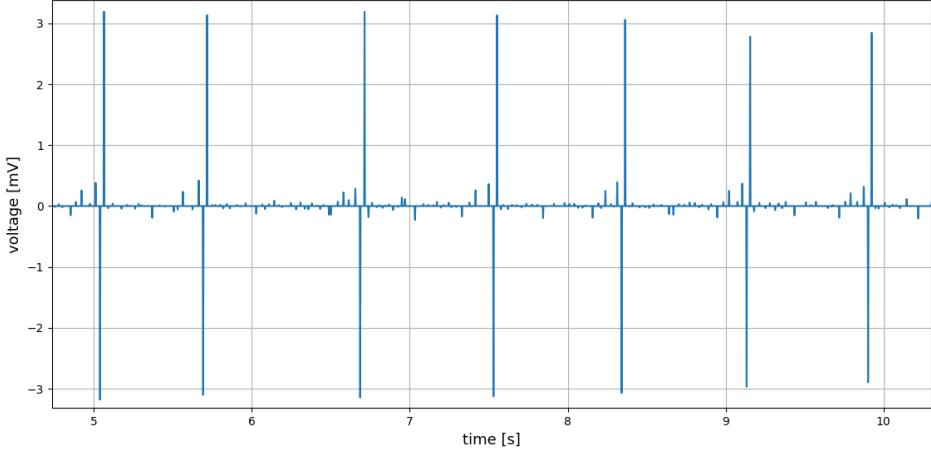


Figure. 3.7: Maximum and minimum values on scale 3.

divided into four parts, let's say, $A1, A2, A3$ and $A4$. To find the average maximum and minimum values, the maximum and minimum values from each part are identified.

$$MAX = \frac{\max(A1) + \max(A2) + \max(A3) + \max(A4)}{4} \quad (3.35)$$

$$MIN = \frac{\min(A1) + \min(A2) + \min(A3) + \min(A4)}{4} \quad (3.36)$$

Not all the extrema are important, therefore, only modulus maxima pair in certain scope can be focused and the rest can be ignored. The threshold are calculated as:

$$Th_1 = \frac{MAX}{2}, Th_2 = \frac{MIN}{2} \quad (3.37)$$

The threshold values can be adjusted. If they are too big, then the missing probability of R peaks would be higher and if they are too low, then the noise or interference can be misdetected as R peaks. The thresholds defined above provided the best result in our experiment. The modulus maxima pair are shown in Figure 3.8.

The best modulus maxima pair corresponds to the R peaks and the value of R peaks lie at the zero-crossing point of modulus maxima pair with a little delay. For compensating the delay, a maximum value can be searched in the window of 20 points to the zero-crossing point to determine the accurate position of R peaks.

To compensate the false R peak detection or misdetection, the distance between the adjacent R peaks is calculated. If the distance is less than 40% of the average RR interval,

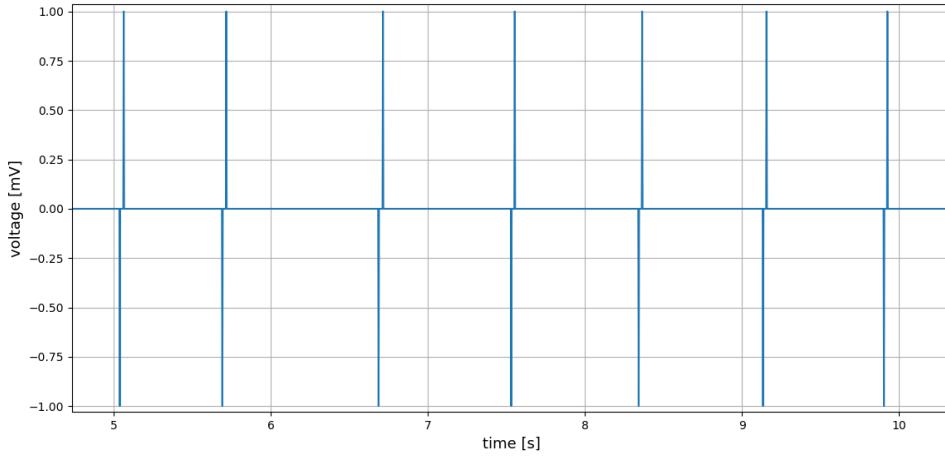


Figure. 3.8: Maximum and minimum values pair for finding the R peaks.

then one of the peaks is misdetected. Therefore, the larger one is kept as R peak and the smaller one is eliminated.

If the distance between adjacent R peaks is greater than 160% of the average RR interval, it is considered as misdetection. Therefore, the threshold values defined in Equation 3.37, are further reduced and R peak is searched again in the adjacent R peaks window or interval.

The detected R-peaks are shown in Figure 3.9. The flowchart for finding the R peaks is shown in Figure 3.10.

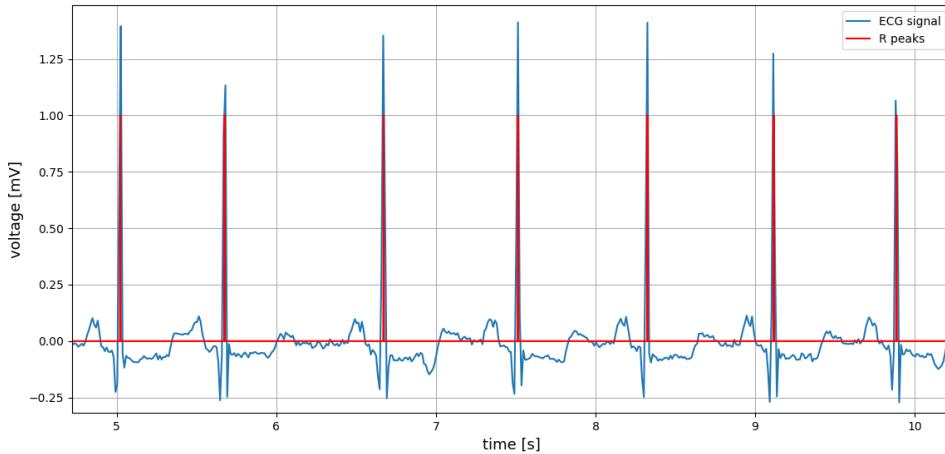


Figure. 3.9: Detected R peaks.

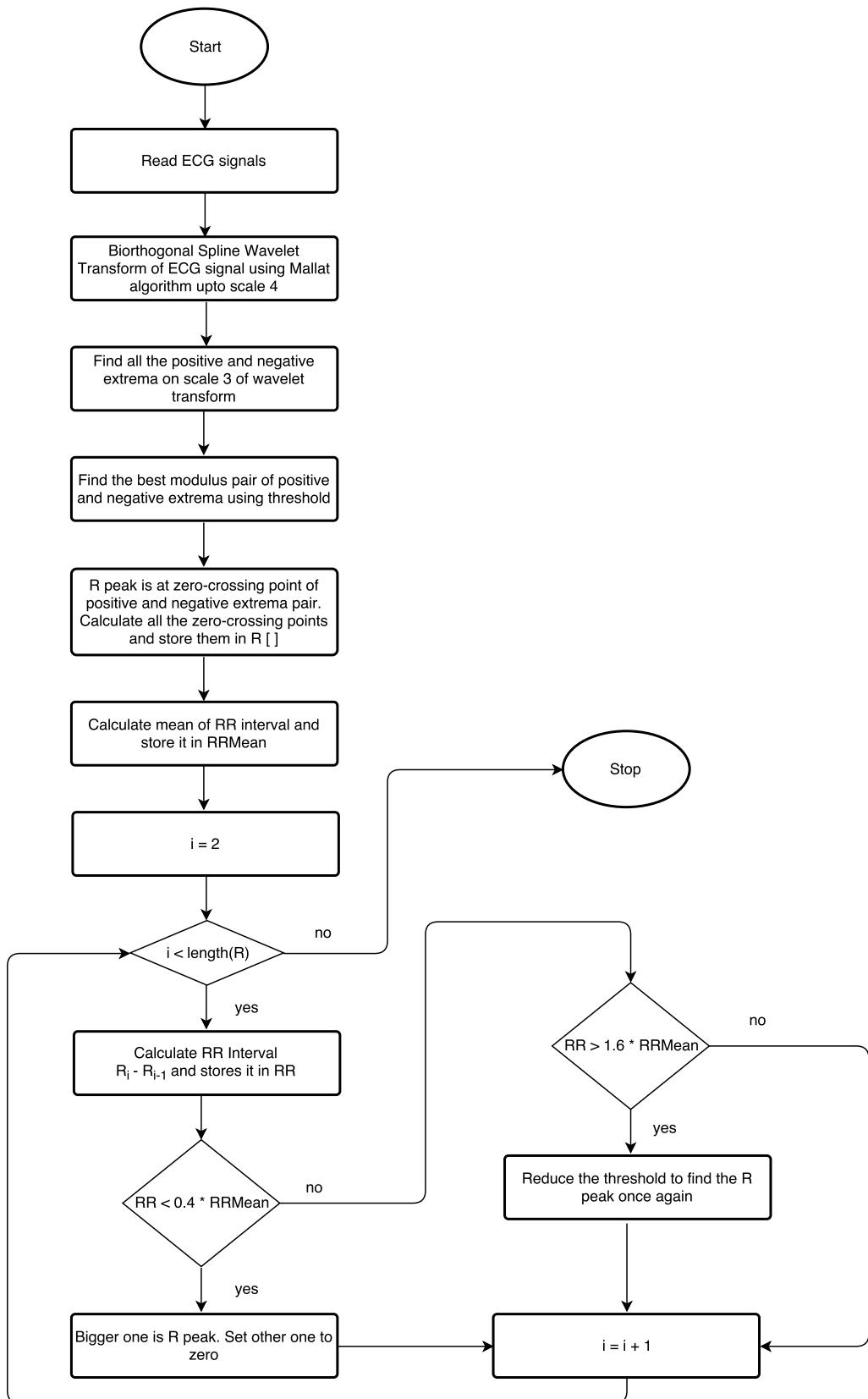


Figure. 3.10: Flowchart for locating R peaks.

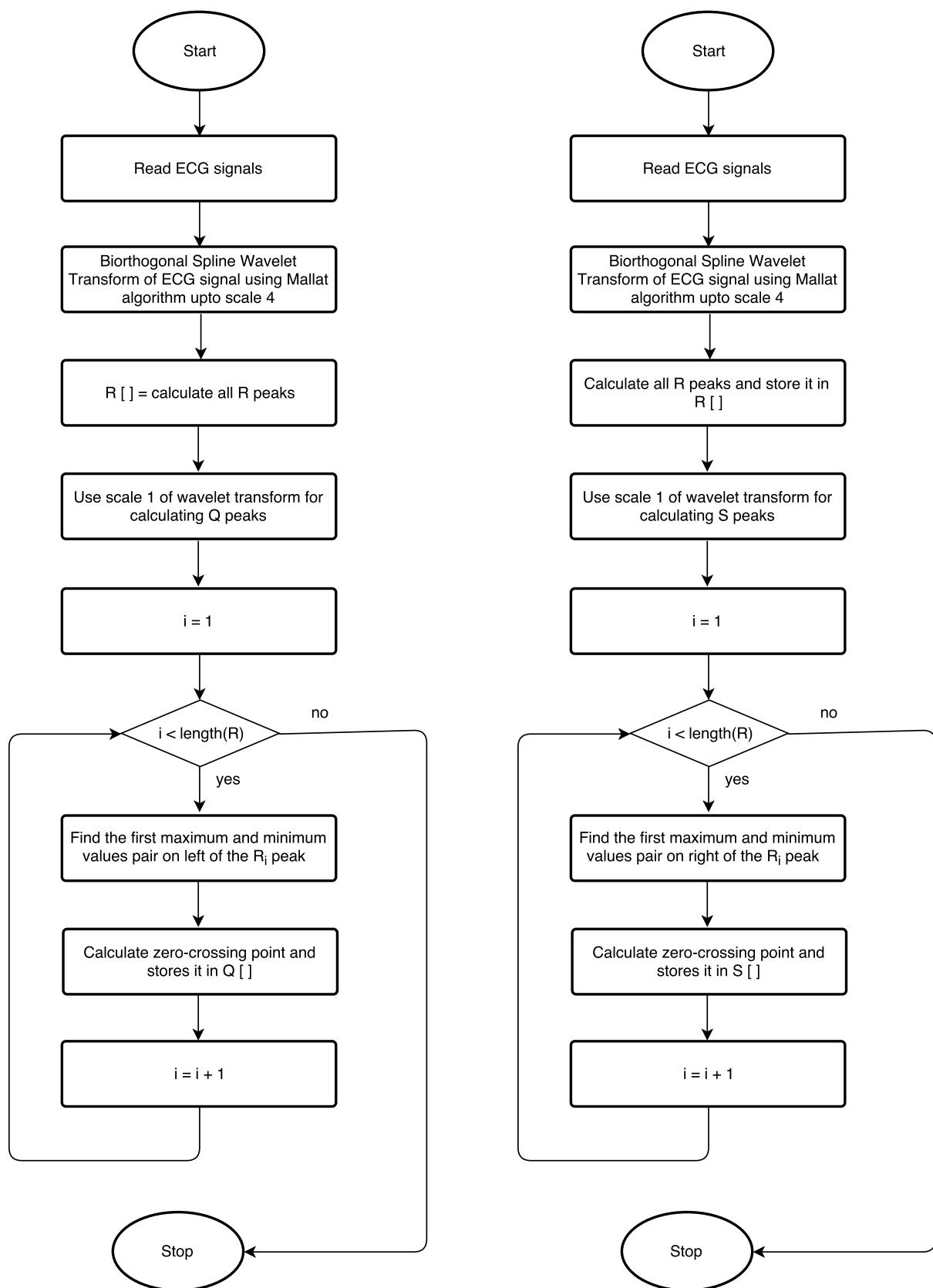


Figure. 3.11: Flowchart for locating Q and S peaks.

Q and S peaks generally are of high frequency. Therefore, their energies mainly lie at the 1st scale. For finding the Q wave, the algorithm starts by looking on the left side of the R wave and finds the first non-zero value i.e. the Q wave. And because of the delay, the minimum value is searched in the window of 10 points to the detected Q wave. The same process is executed for the S wave, but in this case, the direction was on the right side of the R wave. The detected QRS complex can be seen in Figure 3.12.

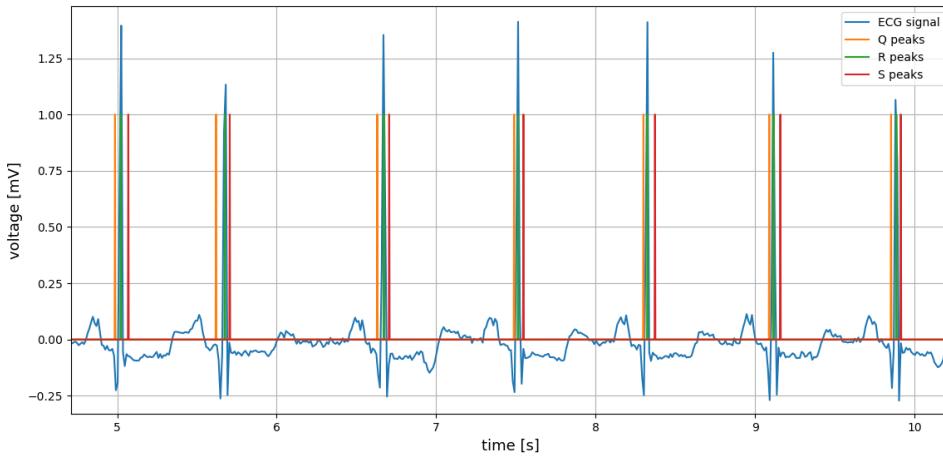


Figure. 3.12: Detected Q, R and S peaks.

After detecting the QRS complex, P and T waves are required to be detected as they are important for the identification of the arrhythmia. P wave occurred before the QRS complex and T wave after the QRS complex. Therefore, they can be detected based on the QRS location.

3.9 P and T Wave Detection

After the detection of QRS complex, the P and T waves are also detected. According to power spectra of ECG signal [TWT84] and Table 3.4, most of the P and T waves energy lies at scale 4. Therefore, the scale 4 is selected to detect P and T waves.

The corresponding modulus maxima pair of P and T waves can be seen in Figure 3.6 at scale 4 before and after the QRS complex. Therefore, P and T waves can be searched in a window before and after the QRS complex. A window size of 100 is used for detecting the P wave and T waves. The P wave is searched within the 33% of the average RR interval to the left of QRS complex and T wave is searched within the 67% to the right of QRS complex. The P and T waves lie at the zero-crossing point of the identified modulus maxima pair in their respective windows. Because of the delay, a maximum value is searched in the

window of 20 points to the zero-crossing point to determine the accurate position of P and T waves.

All detected waves are shown in Figure 3.13. The Flowchart for finding the P and T waves are shown in Figures 3.14 and 3.15 respectively.

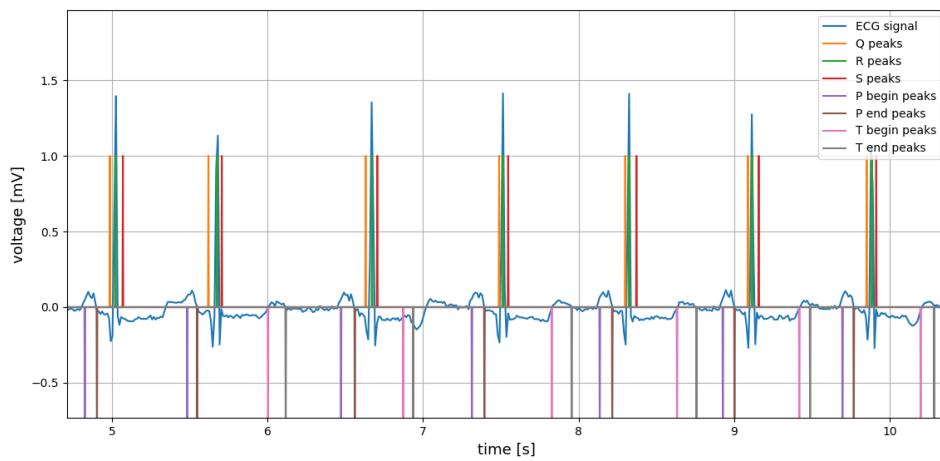


Figure. 3.13: Detected P,Q,R,S and T waves.

3.10 Heart Rate Calculation

The heart rate is calculated by using the RR interval. The formula to calculate the heart rate is:

$$HeartRate = sample_rate \times \frac{60}{RRinterval} \quad (3.38)$$

where *sample_rate* is the sample rate of the ECG signal and 60 is multiplied to convert the heart rate into beats per minute.

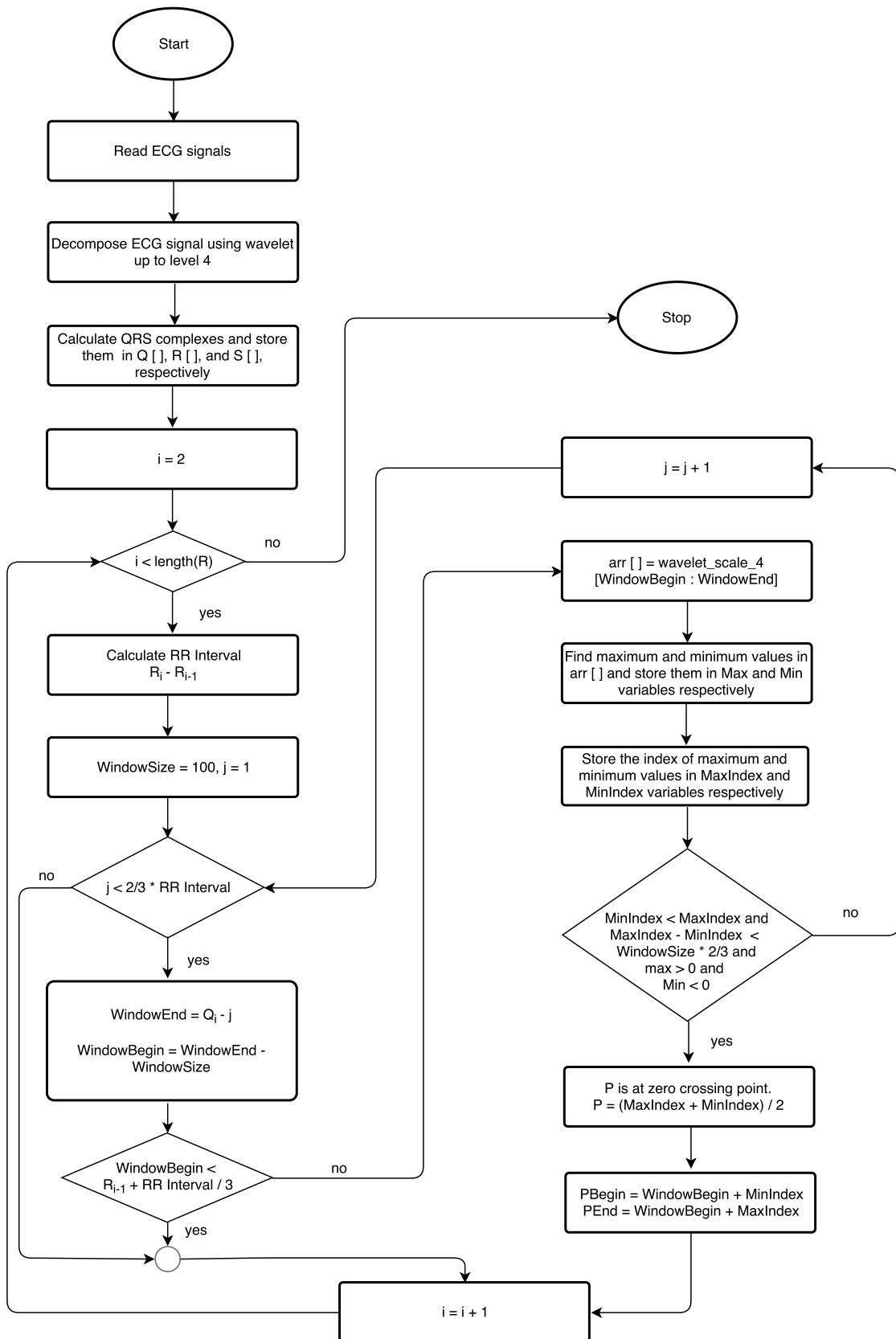


Figure. 3.14: Flowchart for locating P wave.

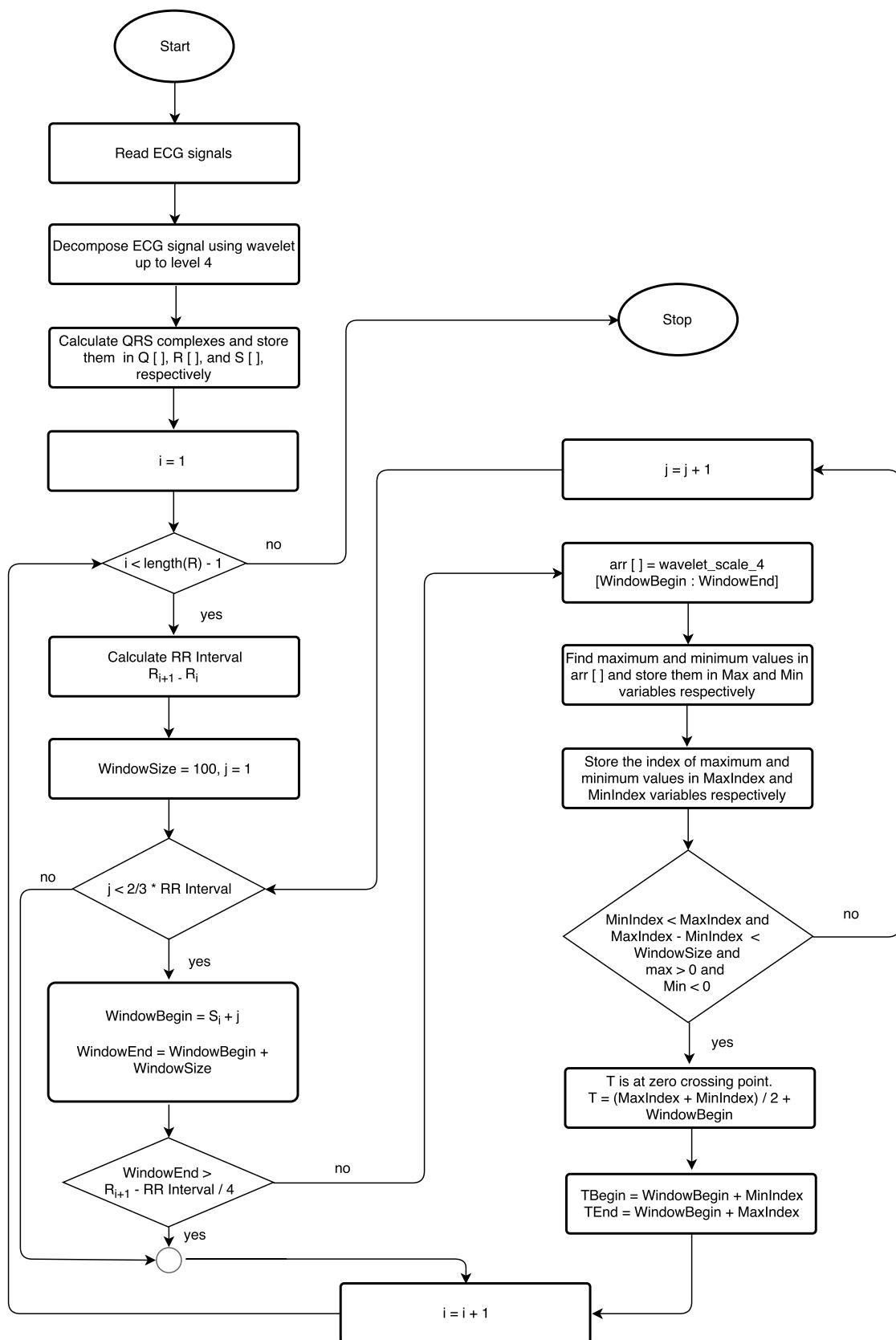


Figure. 3.15: Flowchart for locating T wave.

3.11 Algorithm Execution on ECG Chair Data

The data used in the development of ECG feature extraction algorithm is taken from MIT-BIH dataset and the algorithm works great with that dataset. The original ECG signal from the chair is shown in Figure 3.16 and the extracted features are shown in Figure 3.17.

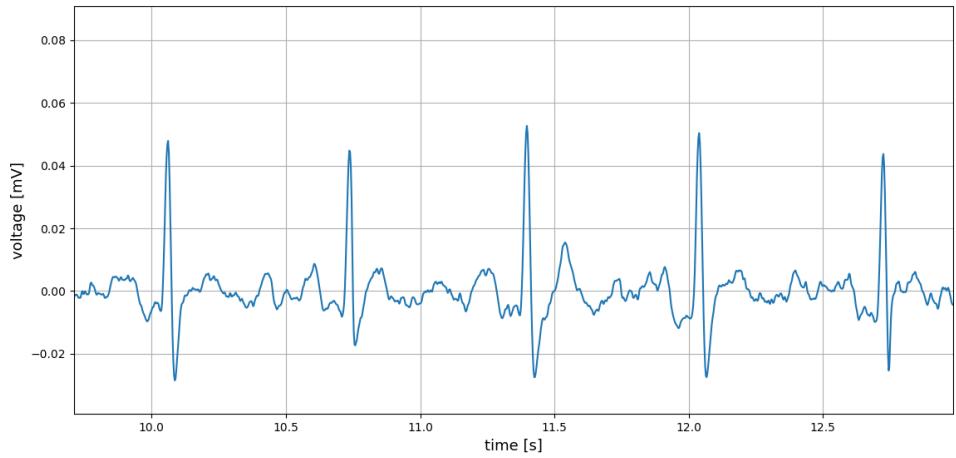


Figure. 3.16: Original ECG signal from the chair.

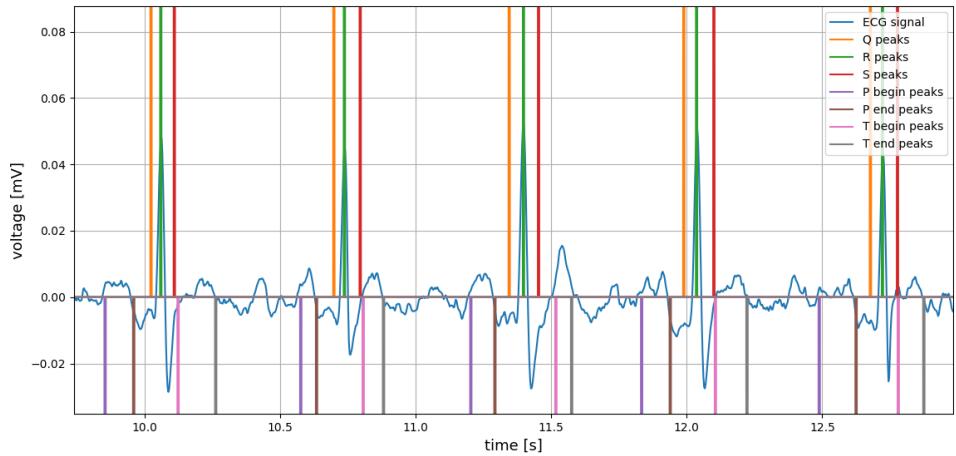


Figure. 3.17: Detected P,Q,R,S and T waves on the chair's ECG signal.

3.12 PPG Signal Processing

A band-pass filter can be used to reduce the baseline drift and high-frequency noise from the PPG signal. A passband of 0.5 Hz to 5 Hz has been used. After applying the band-pass filter, the resulting signal is the denoised PPG signal. The denoised channel 1 signal is shown in Figure 3.18.

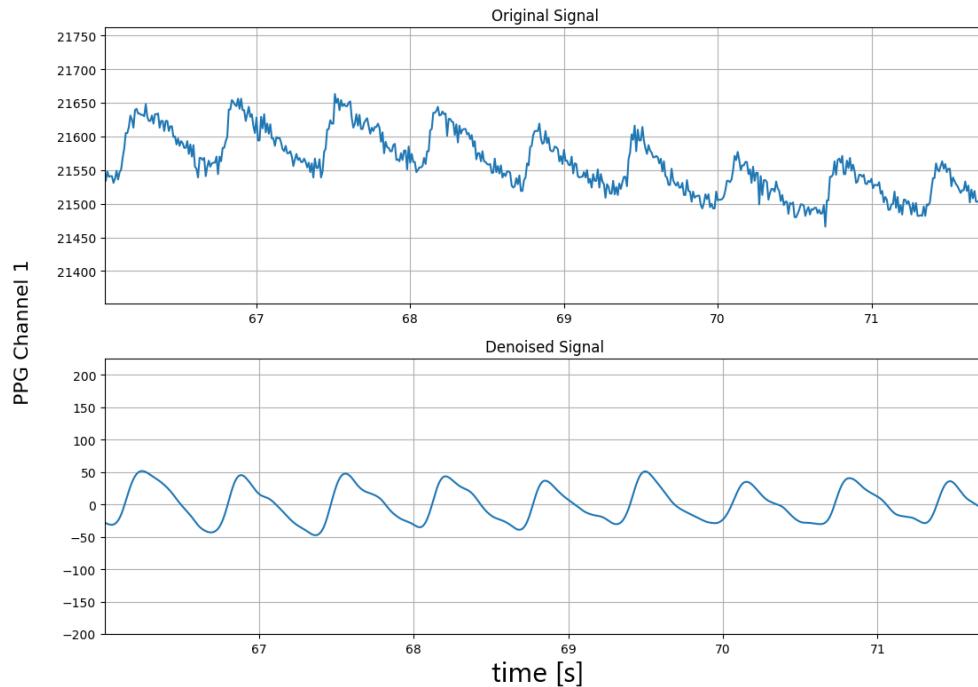


Figure. 3.18: The denoised PPG signal using band-pass filter.

4 Deep Learning

Deep learning is a subfield of machine learning which is in fact, is a subfield of Artificial Intelligence (AI). Deep learning has emerged as one of the most exciting fields of computer science, and it keeps expanding its scope to other domains as well. It has been used in many technologies such as, in medical to identify the diseases, automatic game playing, self-driving cars, image recognition, and natural language processing. The reason why deep learning is successful in many different domains is its ability to understand multiple levels of representation of data. Its mean that, it not only has the ability to classify and predict but also has the ability to learn a different level of complexity. Before diving into deep learning, it is necessary to understand a broader “machine learning”.

4.1 Machine Learning

Machine Learning is a data analysis method [Bis06]. It gives the computer the ability to learn from data without being explicitly instructed. By using different machine learning algorithms, it helps to find hidden insights of data and allow us to build models to make predictions. It can be classified into 2 categories [Bro17].

4.1.1 Supervised Learning

In supervised learning, the labeled data is used to train the models. The labeled data represents that the input and output variables that are known in advance. Thus, a supervised machine learning algorithm is used to come up with a mapping function which maps the input variables to the output variables. Learning is supposed to be stopped when the level of performance reaches the desired result. Supervised learning is generally divided into regression and classification.

- **Regression:** A problem in which the output variable is a category.
- **Classification:** A problem in which the output variable is the real value.

4.1.2 Unsupervised Learning

In unsupervised learning, only the input variables are known and no corresponding output variables are known. Thus, there are no labels given and it is expected from unsupervised machine learning algorithm to find the structure in the data, i.e. finding hidden patterns

to learn more about data. It is different from supervised learning in that, the correct output values are not known. Unsupervised learning is generally divided into clustering and association.

- **Clustering:** Group objects in such a way that the objects, which are similar to each other, placed in the same cluster.
- **Association:** Discover rules that define the large portions of the data such as people who buy product X may buy Y as well.

The objective of machine learning is to analyze the past and present data and predict or make a decision for the future data. In supervised learning, the basic workflow is to build a model, evaluate or tune a model and then deploy it in the production environment where it will make the predictions. The workflow can be seen in Figure 4.1.

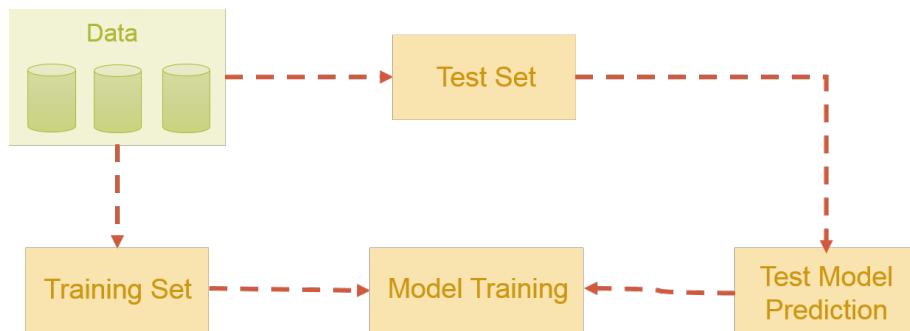


Figure. 4.1: Basic supervised machine learning workflow.

Machine learning is generally powered by a huge amount of data which is generally referred as Big Data. It is generally defined as a too big or complex data which cannot be processed on a single machine. As the data is growing day by day, the new tools are also required to process that big data on multiple machines and extract the useful insights from the data.

One of the problems with the traditional machine learning model is the feature extraction challenge. The model designer or the programmer need to specifically tell the model which features it should consider while making a decision. The model heavily relies on the programmer's understanding of data and this was a huge burden on the programmers. For problems like object recognition and language translation, it was considered as a huge problem.

Deep learning comes into play to solve the problem of feature extraction. They have the capability to focus only on the right features by themselves by understanding as much data as possible, requiring very little input from the programmer. This feature of deep learning models makes it very powerful tool for the current machine learning era.

4.2 Artificial Neural Networks

Artificial neural networks (ANNs) are generally inspired by the biological neural networks that mimic brain functionality [con17a]. These systems generally learned by considering examples instead of specifically define rules for certain situations or cases. An ANN is a network of nodes called artificial neurons which are connected to other neurons using a link called *synapse*. Each neuron gets the input, process the input and pass the output to the next neuron. In the most basic state, an ANN consists of 3 layers:

1. Input Layer
2. Hidden Layer
3. Output Layer

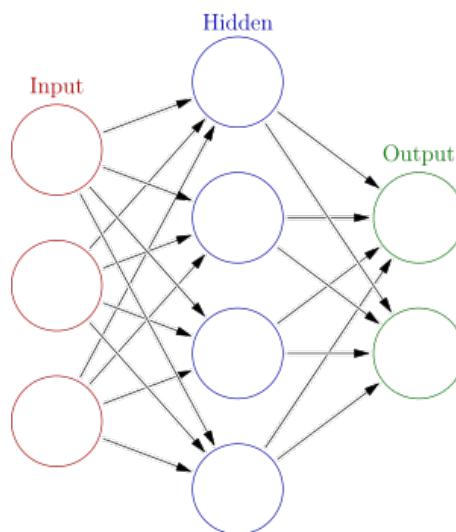


Figure. 4.2: An artificial neural network with its 3 basic layers [con17a].

4.2.1 Artificial Neuron

An artificial neuron is the most basic unit of ANN. It takes inputs and produces an output. Generally, the inputs are multiplied by some weights to specify which inputs are more important. The higher the value of weight is, the more important it is. The inputs are shown as a , b , and c , and weights as w_1 , w_2 and w_3 in Figure 4.3. After then the products are summed together and passed to the activation function. So, if the summed value is greater than the threshold value of the activation function, the output is produced or in other terms, the neuron fired else no output is produced and neuron does not fire.

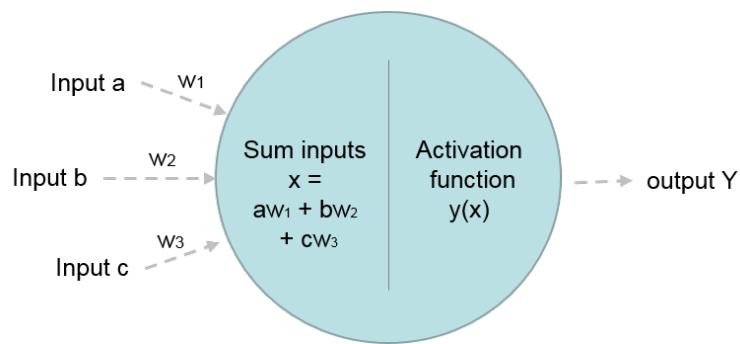


Figure. 4.3: A single artificial neuron.

Artificial neurons adjust the weights as the learning proceeds and the process of finding weights is known as learning. ANN considers many different examples and finds the best possible combination of weights to provide the most accurate results. There are many other parameters involved as well to find a good combination of weights.

4.2.2 Activation Function

A function that takes an input and produces output based on threshold value is known as activation function [ujj17]. There are many activation functions available. Few of them are:

Sigmoid

It takes a real value input and scales it to the range of 0 to 1. It is also known as the logistic function. It is represented as:

$$y = \frac{1}{1 + e^{-x}} \quad (4.1)$$

Another variation of the sigmoid function is softmax function which is used for multiclass classification.

Tanh

It takes a real value input and scales it to the range of -1 to 1. It is also a sigmoidal function as it also takes s-shaped.

ReLU

It stands for Rectified Linear Unit. It is the most used activation function with convolutional neural networks[ujj17]. It takes the real value input and all negative values are mapped to zero. It is represented as:

$$f(x) = \max(0, x)$$

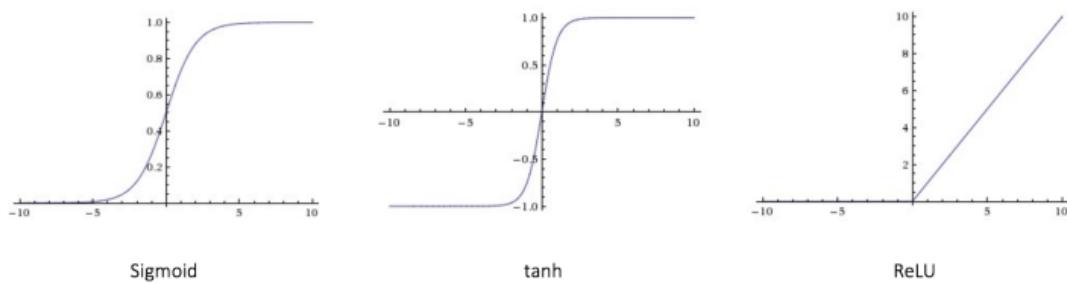


Figure. 4.4: Activation functions [ujj17].

The graph of all the activation functions defined above can be seen in Figure 4.4.

4.3 Convolutional Neural Network

Convolutional neural network (CNN) is a class of deep neural network which uses multilayer perceptrons. It consists of an input layer, an output layer, and multiple hidden layers. The hidden layers can be convolutional, pooling or fully connected layer. An example of CNN is shown in Figure 4.5.

4.3.1 Convolutional Layer

In CNN, the first layer is always the convolutional layer. This layer applies a convolutional operation to the input and passes the output to the next layer. A filter (or sometimes referred as a kernel) is used which slides over all the area of the input and extract the features from it. The region where the filter is being applied at any instant of time is called as a receptive field. As the filter slides or convolves over the input signal, it multiples the filter with the original signal values (which can also be referred as element-wise multiplication) which results in a single value. The important point to note here is that this single result

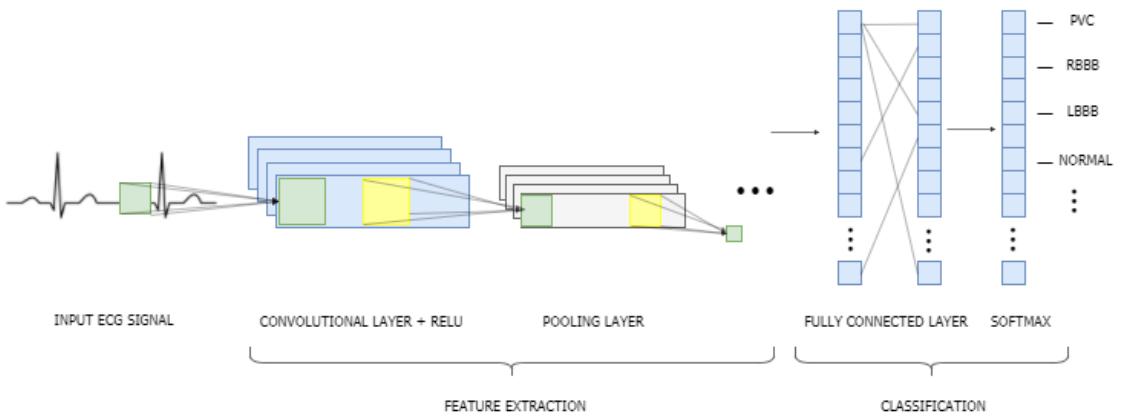


Figure. 4.5: Convolutional neural network example.

is just from a single receptive field. Therefore, the same operation will be carried out on other receptive fields by sliding the filter over the different areas of the input signal. The filter can slide for any unit and this sliding unit is known as stride. Every unique area of the input signal will produce an output and the combined output together is called as feature map or an activation map. If we assume that, the input size is 32x32, the filter size is 5x5 and the stride is 1 unit, then the size of the activation map will be 28x28. In a 2D array, the filter can be moved in both directions.

4.3.2 Pooling Layer

Pooling layer is used to downsampled the convolutional layer output. There are several pooling options, for example, average pooling and L2-norm pooling, but max pooling is the most popular. This layer basically takes a filter of size 2x2 with the same stride size. It then applies the filter to the part of input and output the maximum number in that region. The same process is applied to the different sub-region by sliding the filter all over the input. By convolving the filter around the input, it drastically reduces the spatial size of the input. The example of how pooling layer work is shown in Figure 4.6.

4.3.3 Fully Connected Layer

In a fully connected layer, the neurons of one layer are connected to all the neurons of the second layer, as shown in Figure 4.5. This layer can be seen in the regular neural network as well. The softmax function is applied to the output of the second layer to produce the probabilities for the classification of the input.

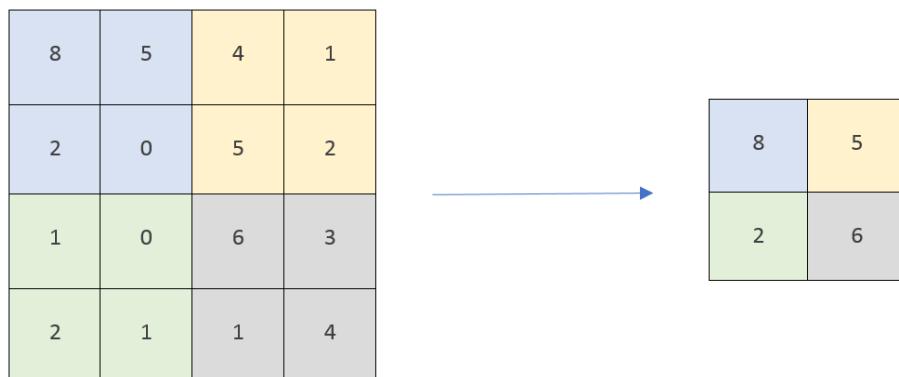


Figure. 4.6: Max pooling layer example.

4.4 Keras

Keras is an open source artificial neural network library written in python. It is very powerful and easy to use library to develop neural networks. It has a capability to run on top of TensorFlow, Microsoft Cognitive Toolkit (CNTK), or Theano, can run on both CPU and GPU. Before Keras, it was really time-consuming to develop a network on TensorFlow or Theano and the aim to develop Keras is to make it easy and fast to develop neural networks. Keras supports both convolutional neural networks and recurrent neural network, as well as a combination of both.

The model starts by defining the model as sequential using a `Sequential()`, which is a linear stack of layers. Then the layers are added into the model using `add()` method. Keras supports almost all kinds of layers. The input dimensions are needed to specify when the layers are added to the model. Once the layers are added, the model is compiled by using the `compile()` method, which additionally needs 3 arguments.

- Optimizer: It is used to optimize the neural network. Examples of optimizer are RSE, Adagrad and Adam.
- Loss Function: This is the value that model tries to minimize to calculate the error. For example, categorical cross-entropy and MSE.
- Metrics: It can be any existing metric or a custom defined metric function. But for classification problems `metrics=['accuracy']` is recommended.

The model is trained by using the `fit()` method. This method lets the model iterate over the data and finds the most optimal neural network for the given data.

Keras has been used for training the CNN to identify cardiac arrhythmias.

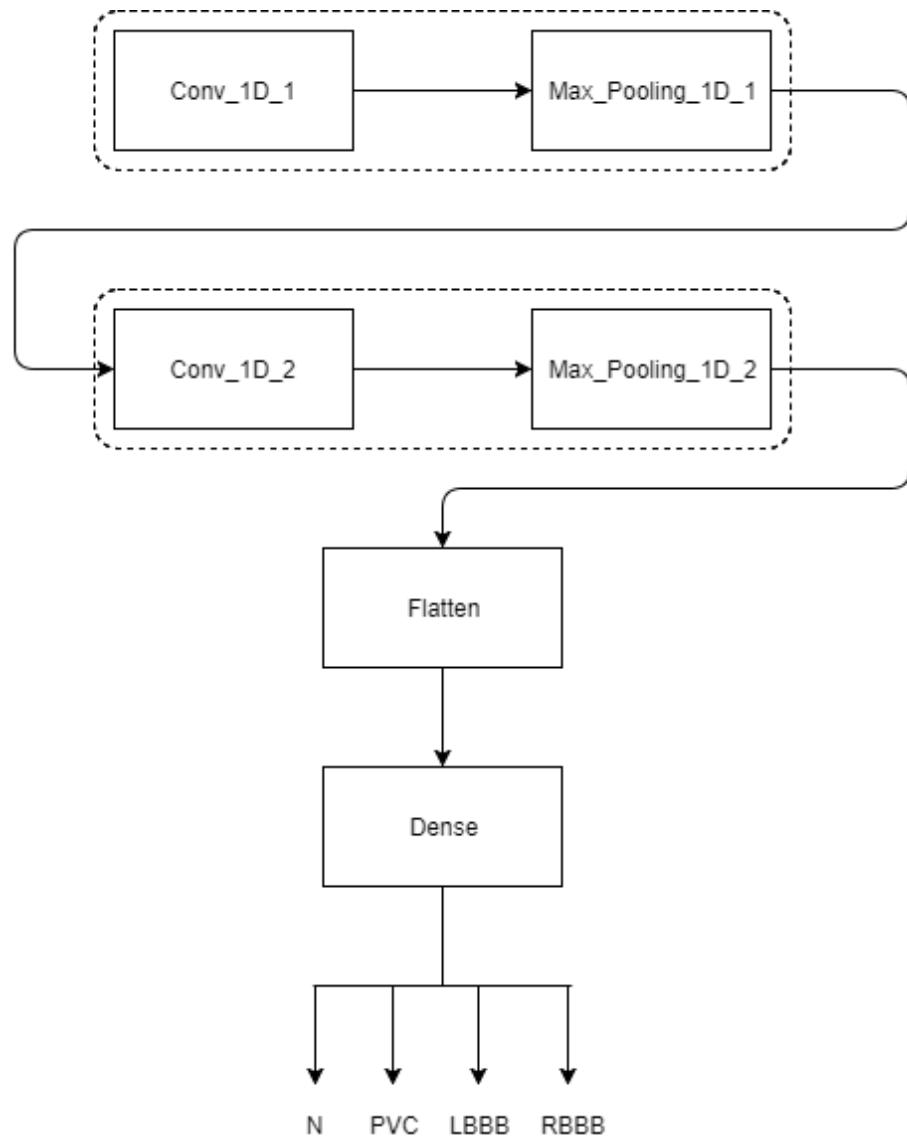


Figure. 4.7: A 6-layer Convolutional Neural Network model for the identification of cardiac arrhythmia.

4.5 Convolutional Neural Network for the Identification of Cardiac Arrhythmia

A 6-layer Convolutional Neural Network (CNN) has been trained for the identification of an arrhythmia in an ECG signal. The trained model can detect 3 different kinds of arrhythmia namely:

1. Normal
2. Left bundle branch block (LBBB)
3. Right bundle branch block (RBBB)
4. Premature ventricular contraction (PVC)

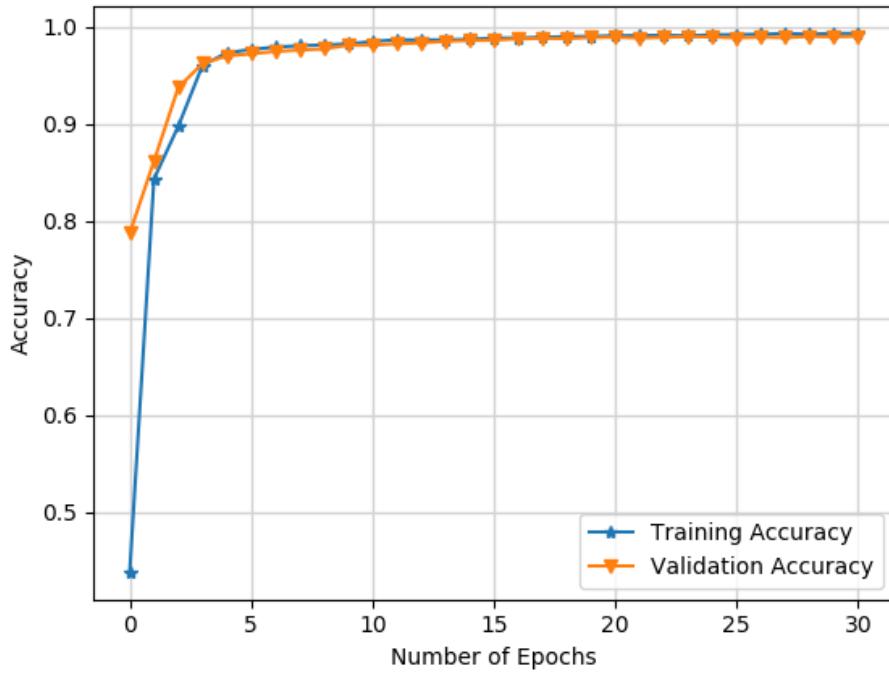
The CNN layers are shown in Figure 4.7.

Layer (type)	Output Shape	Param #
conv1d_1 (Conv1D)	(None, 300, 64)	256
max_pooling1d_1 (MaxPooling1)	(None, 150, 64)	0
conv1d_2 (Conv1D)	(None, 150, 32)	6176
max_pooling1d_2 (MaxPooling1)	(None, 75, 32)	0
flatten_1 (Flatten)	(None, 2400)	0
dense_1 (Dense)	(None, 4)	9604
<hr/>		
Total params: 16,036		
Trainable params: 16,036		
Non-trainable params: 0		

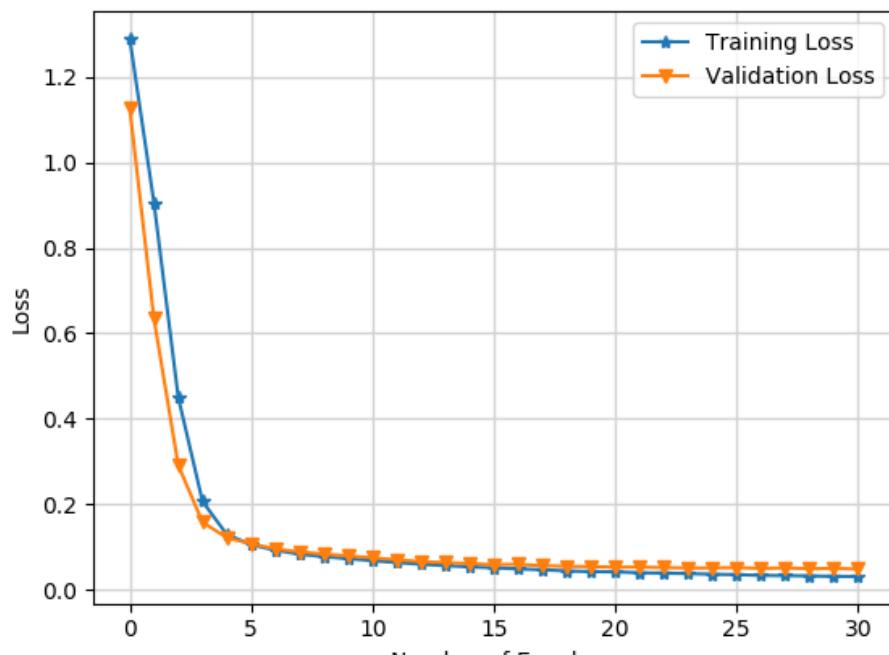
Figure. 4.8: Layers used in the model and number of parameters to be optimized.

4.5.1 Layers Explanation

The 1st convolutional layer (Conv_1D_1) consists of 64 filters, whereas, the 2nd convolutional layer (Conv_1D_2) consists of 32 filters with a kernel of size 3. The model need to optimize total of 16,036 params to find an optimal CNN, as shown in Figure 4.8. For both convolutional layers, Relu has been used as an activation function each followed by a max-pooling layer. The batch size of 1,000 was used for the training, along with the Adam algorithm to optimize the CNN. Since the model is trained for performing the classification



(a)



(b)

Figure. 4.9: (a) Training and validation accuracy results; and, (b) training and validation loss of CNN model for 30 iterations.

Tab. 4.1: Training data.

Type	Count
Normal	3,352
LBBB	2,641
RBBB	2,498
PVC	2,507
Total	10,998

of an ECG signal, therefore, categorical cross-entropy loss function is used for calculating the loss of training and validation. After performing the convolutions, the flattening and dense layer has been used, followed by a softmax activation function to produce the final probabilities with 4 different classes.

The model is trained and tested for the several numbers of iterations ranging from 10 to 100. The best model was found after 20 iterations. After that, the model remained stable with the slight improvement in the training as well as in the validation accuracy. The training and validation accuracy can be seen in Figure 4.9.

4.5.2 Results

The CNN model has achieved an accuracy of 99.2% on MIT-BIH dataset. Total 16,415 ECG signals were extracted from the MIT-BIH dataset, out of which 10,998 (67% of the total signals) were used for the training, and the remaining 5,417 (33% of the total signals) were used for the testing of the model. The types count for training and testing data can be seen in Tables 4.1 and 4.2 respectively.

During the testing of the model, 42 ECG signals were classified wrong out of 5,417 ECG signals. The false predicted results can be seen in Table 4.3. In the table, $a \rightarrow b$ represents that a was classified as b .

Tab. 4.2: Test data.

Type	Count
Normal	1,648
LBBB	1,308
RBBB	1,285
PVC	1,176
Total	5,417

Tab. 4.3: False prediction counts.

False Prediction	Count
$0 \rightarrow 1$	1
$0 \rightarrow 2$	1
$0 \rightarrow 3$	3
$1 \rightarrow 3$	6
$2 \rightarrow 0$	5
$2 \rightarrow 3$	3
$3 \rightarrow 0$	9
$3 \rightarrow 1$	12
$3 \rightarrow 2$	2
Total	42

5 Visualization and System Architecture

“A picture is worth a thousands words.”

Raw numbers to the users do not make sense and therefore, require necessary tools to display the results. Visualization allows us to see the broader aspects of complex data by showing the data in graphical formats. It really helps in capturing the user’s attention and engage him throughout the process. Complex data that could easily be ignored, can still be recognized and captured the attention of the user in a graphical reports [Iye17].

The visualization tool Grafana has been set up for displaying the real-time data received from the sensors. All devices send the data in real-time, which first get stored in Influxdb and then Grafana tool loads the data from there and display it on the graph.

5.1 Grafana

Grafana is an open source real-time visualization tool for analytics and monitoring. It is one of the best tools for time series analytics, therefore, it has been used for visualizing the real-time data from sensor devices. It can be used for any kind of application analytics, for example, industrial sensors, home automation, hospitals, and weather reports. It can connect to many data sources and pull data from there to do the analytics and the visualization. The most commonly used data sources these days are:

- Elasticsearch
- InfluxDB
- Graphite
- Prometheus

It allows us to connect to these data sources on just few click, which makes it very convenient. Multiple dashboards can be created in Grafana to view different dimensions of the data. It also provides multiple tools for creating graphs in different fashion and styles, which can be added to dashboards.

5.2 InfluxDB

Since the sensor data is always time critical, therefore, a time series database is required for storing the data. InfluxDB is one of the best time series database available. Therefore, it has been chosen for storing the sensors data.

It is very easy to install and manage and does not require other dependencies to run. It also provides an HTTP/HTPPS interface to read and write data from the database. The retention policy can be set on the database to manage space conveniently. The basic terms in InfluxDB are:

- Database name
- Measurement (same as table name in traditional databases)
- Tags (to filter data)
- Fields (actual data values)

The fields are generally used as a key-value pair, with a timestamp field. Only one point can be stored at any specific timestamp. The precision of a single field can be in second, millisecond and nanosecond. If the field does not contain a timestamp field, then the InfluxDB will generate a timestamp automatically.

5.3 Setting up Grafana with InfluxDB

Another reason for choosing the InfluxDB is that it is really easy to configure Grafana for using InfluxDB as a data source. The setup can be done in just a few minutes. Before starting, a database should be set up on InfluxDB. Once the database is ready, starts the Grafana server. Generally, it runs on port 3000 but because of ports conflict, it is recommended to change the port to some other address by editing the *conf/sample.ini* file.

Once the server is started, browse *localhost:3000* to open the Grafana web interface. Select an option to add a new data source and configure the InfluxDB data source as shown in Figure 5.1. A database name, an IP address where the database is running, a username and a password are required to set it up.

Once the data source is set up, a dashboard is created to add the graphs and panels where the data is visualized. Multiple types of panels are available by default such as table, graphs, text, single stat and alert list.

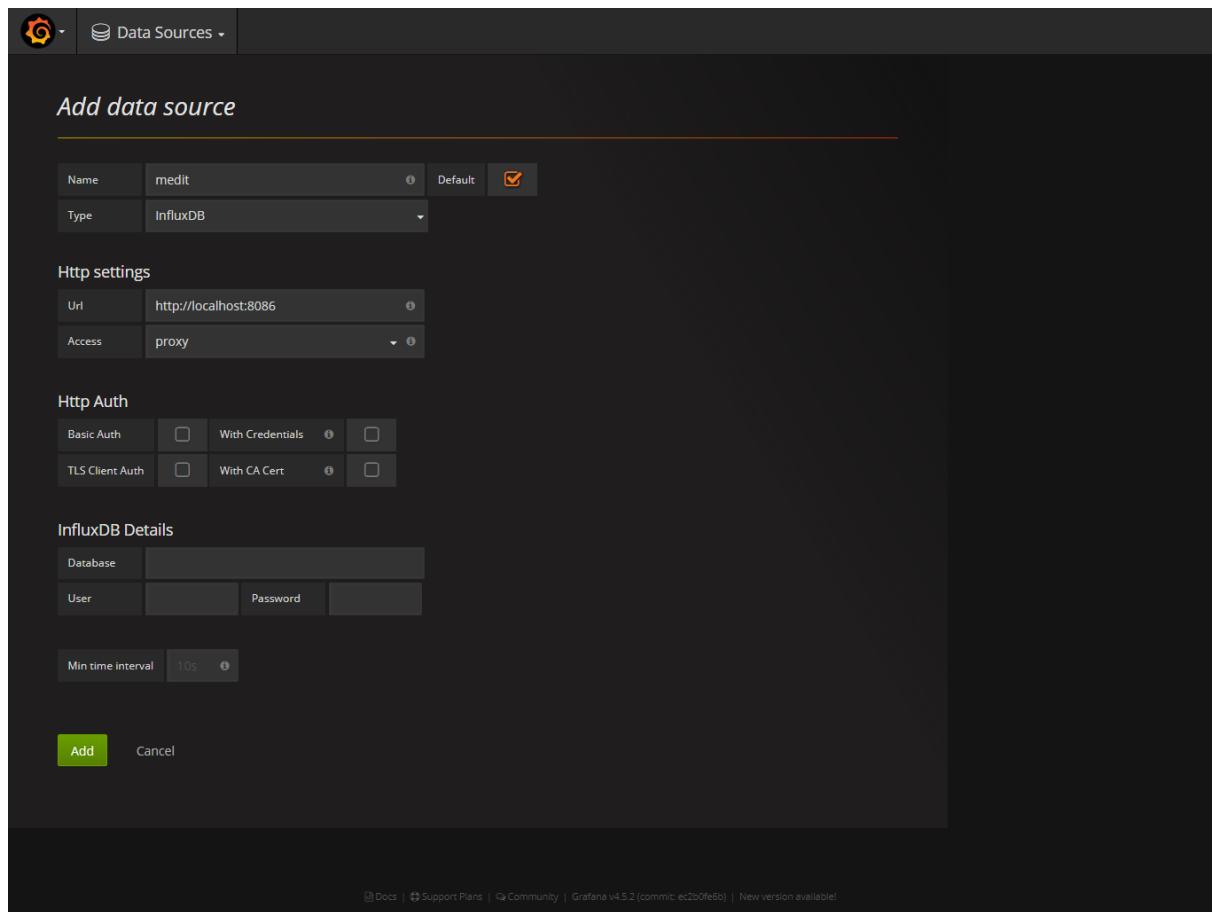


Figure. 5.1: InfluxDB data source setup in Grafana.

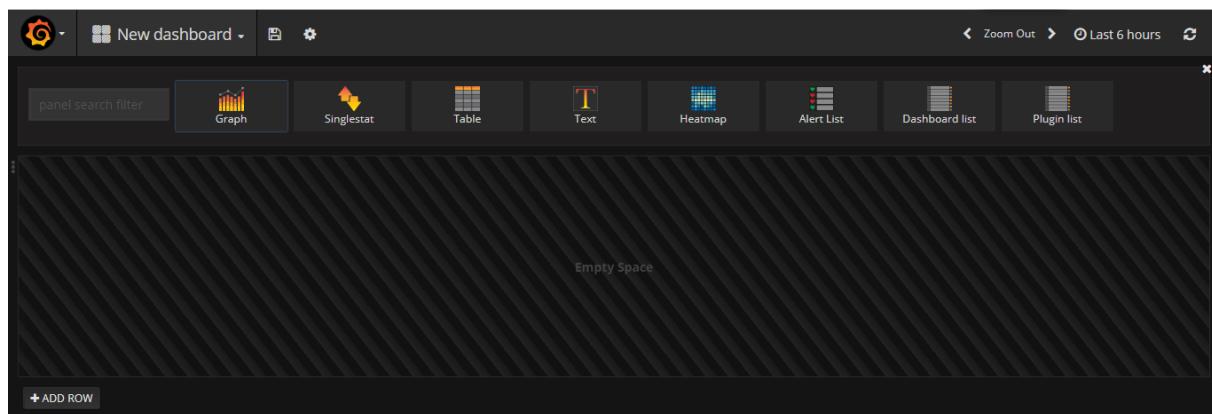


Figure. 5.2: Dashboard setup.

When a graph is added to a dashboard, a data source is needed to be defined for the graph. That can be done by selecting the graph and edit it. The interface for setting up the data source for the graph can be seen in Figure 5.3. A very user-friendly interface is available where one can define a query for the graph. A query can be build just by selecting the option from drop down. It will contain all the information from the database that is set up in the data source, that shown in Figure 5.1. A user simply needs to select the data source, then from the measurements select the specific measurement that wants to be chosen. One measurement can have multiple fields, therefore, select a specific field, which needs to be displayed on a graph. Once all the steps are done, the panel is ready to display the information from the InfluxDB.

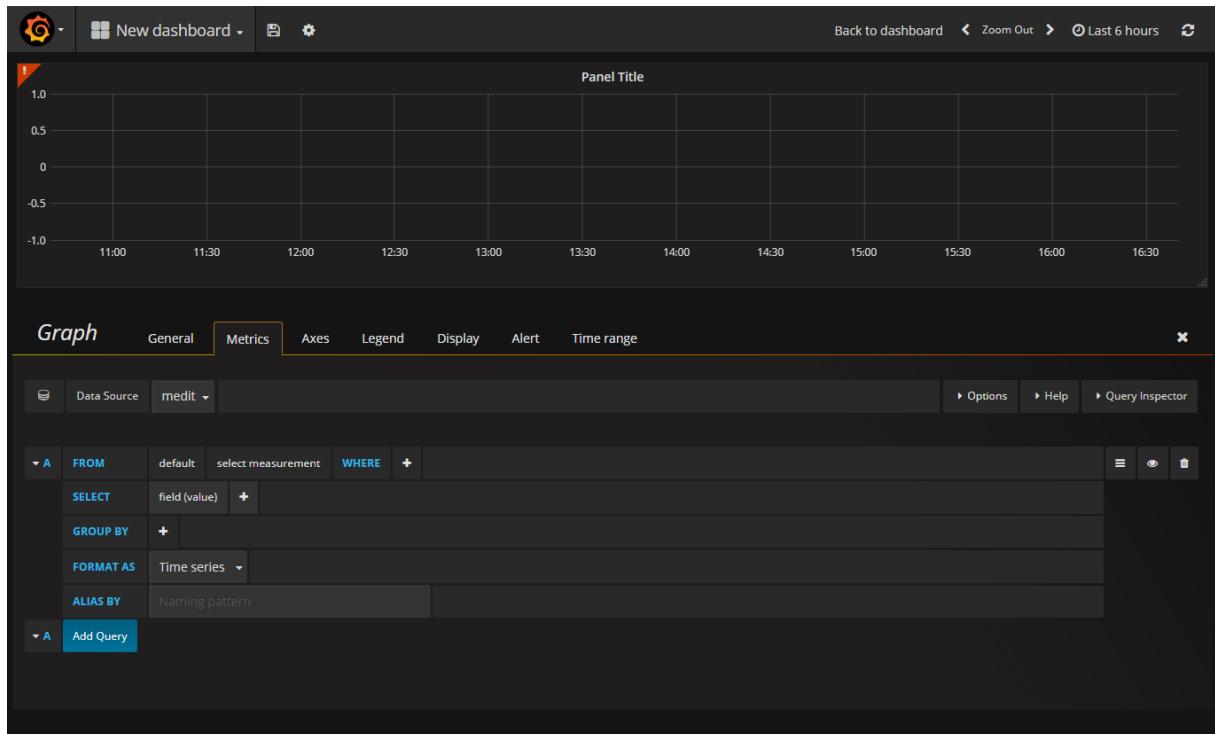


Figure. 5.3: Interface to add query for the panel.

5.4 Lambda Architecture for Big Data Processing

The amount of data being processed nowadays is enormous, and this is often coined as Big Data. Big data is everywhere in the form of weblogs, events, social networks, and sensors data, and these all are generating around petabytes of data each day. And because of this humongous amount of data, the traditional tools and storage technologies are unable to handle and cope with it. Therefore, this has led to a technology phase shift how we keep and manage our data, and to the development of Advance Analytics solutions.

Lambda Architecture [BSK16] [Kre17] [KMM⁺15] [Hau17] is getting involved in machine

learning and data science applications day by day by enabling the real-time data processing and analytics without using the traditional ETL approach. It is designed to address the fault-tolerance, scalability and robustness issues of big data systems. Moreover, Lambda Architecture also ensures the low-latency and accuracy of the result. It combines the power of stream processing with batch processing to provide such kind of system. Lambda architecture, as shown in Figure 5.4, consists of 3 main components:

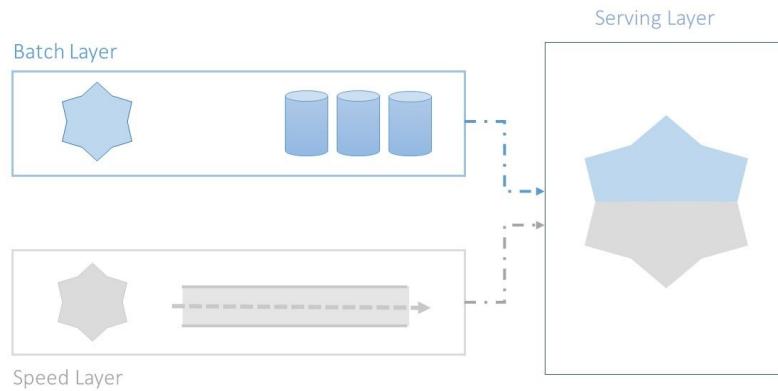


Figure. 5.4: Lambda architecture.

5.4.1 Batch Layer

The Batch layer holds all of the master data, which is stored in Apache Hadoop. This data is kept in its original state, untouched and in an immutable manner. This data is processed and generate the batch views, which are served in the serving layer. The batch view provides the most accurate results from the data using any of the available distributed platform tools.

5.4.2 Speed Layer

Typically, it is known that a Hadoop processing frameworks are slow and take a lot of time. To cope up with this problem, Lambda Architecture introduces a speed layer. Speed layer processes the real-time stream and generates the real-time views of a short time frame, which are then served in a serving layer along with batch views. The main point to notice here is that the speed layer data is temporal in nature, i.e. it can only store that much amount of data that could be kept in the memory. And it is deleted as soon as one batch process is completed.

5.4.3 Serving Layer

This is a place where the final results and data are visualized. Serving layer provides an interface, which integrates real-time views with batch views and unified them together. An accurate view of data will be presented by the batch layer, whereas the fresh view of the data will be presented by the speed layer. It also supports ad-hoc queries, which are optimized for low-latency. Technologies that can be used in this layers are Cassandra and HBase.

The data source provides the data, which is streamed in the stream layer and at the same time to the batch layer. A batch layer holds the data for a long time and stream layer processes the stream in a window of short time and then provide the calculated result. The serving layer combines the data received from the batch layer, more specifically the batch views, and the data received from the speed layer, and allow them to query from a single interface. The advantage of the Lambda Architecture is that if one layer is down, the other layer can be used to make the system available. For example, in Figure ??, if a speed layer is down, which of course can happen in the production environment, The batch layer can be used to compensate the failure.

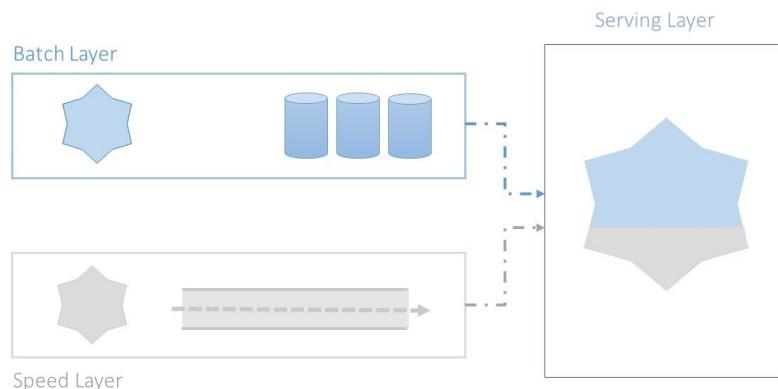


Figure. 5.5: Using more data from serving layer in case of speed layer having a problem.

The reason why it is called lambda architecture is that the lambda symbol splits into two parts, which in case of this architecture represents the batch layer and the speed layer.

Typically, the data stream is implemented using a publish-subscribe messaging system such as Kafka, which can easily scale for high-velocity data ingestion. It can be thought of as a place where publishers publish their data and the consumers read data from it. This is different from the traditional queue as it keeps the data until we let it hold. In contrast, the data is removed from the queue, once it is delivered to the appropriate consumer of the data. It uses topics to publish and to subscribe to the data. Every Kafka server is



Figure. 5.6: Lambda symbol representing the lambda architecture.

known as a broker and since it is a distributed system, therefore, there can be more than one broker. The more the number of brokers is, the higher the availability of the system is. The advantage of having Kafka is that it can handle many different forms of data, such as sensors data, application events, server logs, and social network events. Kafka is very fast despite having a heavy load.

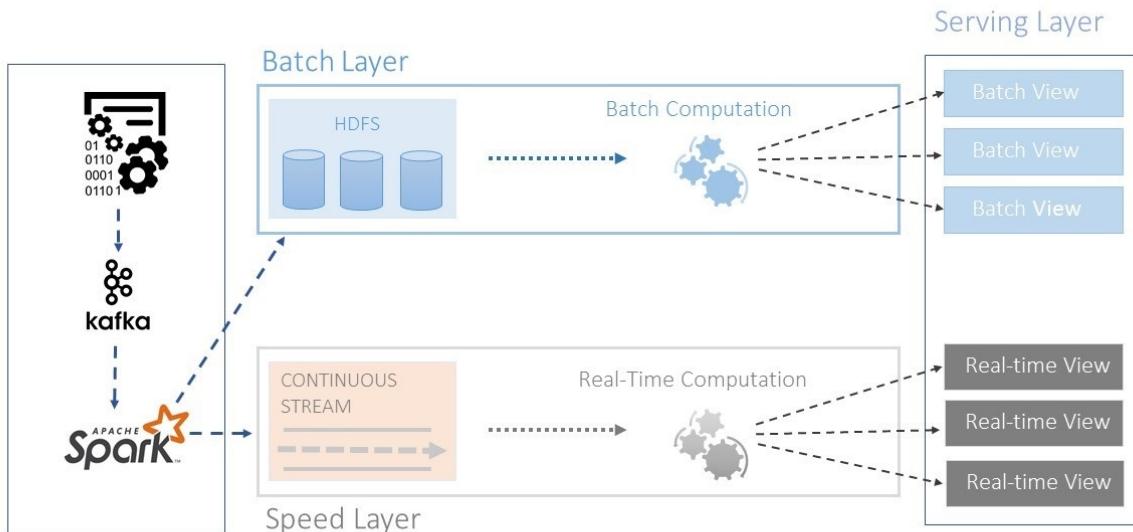


Figure. 5.7: Implementation of lambda architecture.

The implementation of lambda architecture is shown in Figure 5.7. The data is directly published from the data source to Kafka based on some topic. Data from multiple data sources can be collected via Kafka based on different topic names. Once the data is fed into the Kafka, the corresponding consumers will read the data from Kafka using the topic name. In this architecture, Apache Spark can be used in both the batch layer and the speed

layer.

Apache Spark is a large-scale data processing tool, which runs 100 times faster than the Hadoop's MapReduce by caching the data objects in the memory. Thus, it is a strong candidate to replace the MapReduce. It also creates a lineage graph. So, in case of failure, it can do the computations again and go back to the last state of the data. These are two of the fundamental things that Resilient Distributed Data Set (RDD) is all about. It is specially designed to schedule and execute a large amount of data. Apache Spark does not only provide the batch processing, but it also comes with real-time stream processing, machine learning tools, graph processing, Spark SQL, Spark R and complex analytics. It is designed to run everywhere, such as it can run on Hadoop YARN, Apache Mesos or standalone as well. Apache Spark offers an interface for several languages to write the applications. The available languages are Java, Scala, Python, and R.

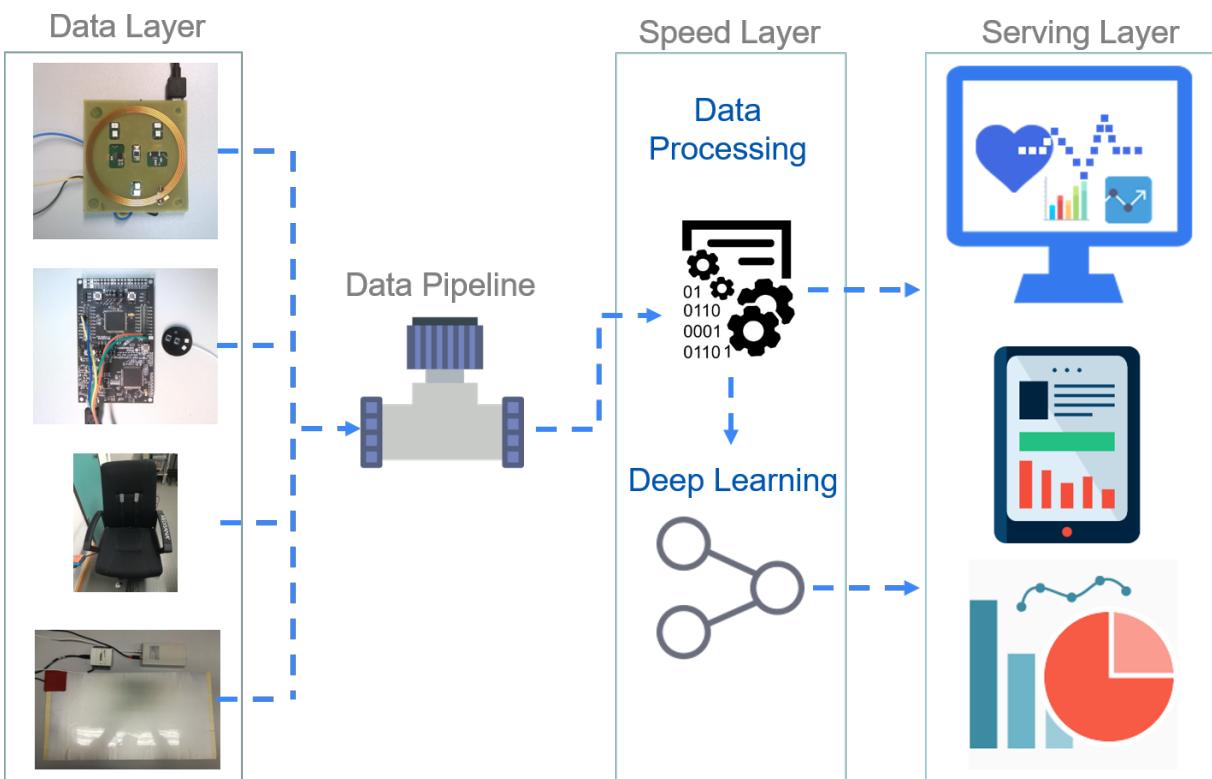


Figure. 5.8: System architecture for the sensors.

5.5 System Architecture

A system has been implemented to collect, process and visualize the data of all the sensors based on the lambda architecture defined in Section 5.4. But the architecture has been modified because of the lack of enough resources to run all the big data tools. The modified

system architecture can be seen in Figure 5.8. The python language has been used for the system implementation. Moreover, big data tools such as Kafka and Apache Spark have been removed from the architecture.

The architecture contains 3 main layers:

5.5.1 Data Layer

The data layer contains all the sensors, which generate data in real-time. The data is collected via data pipeline based on their protocol and submit to the speed layer. The data is also immediately stored in InfluxDB in order to have the original state of the data.

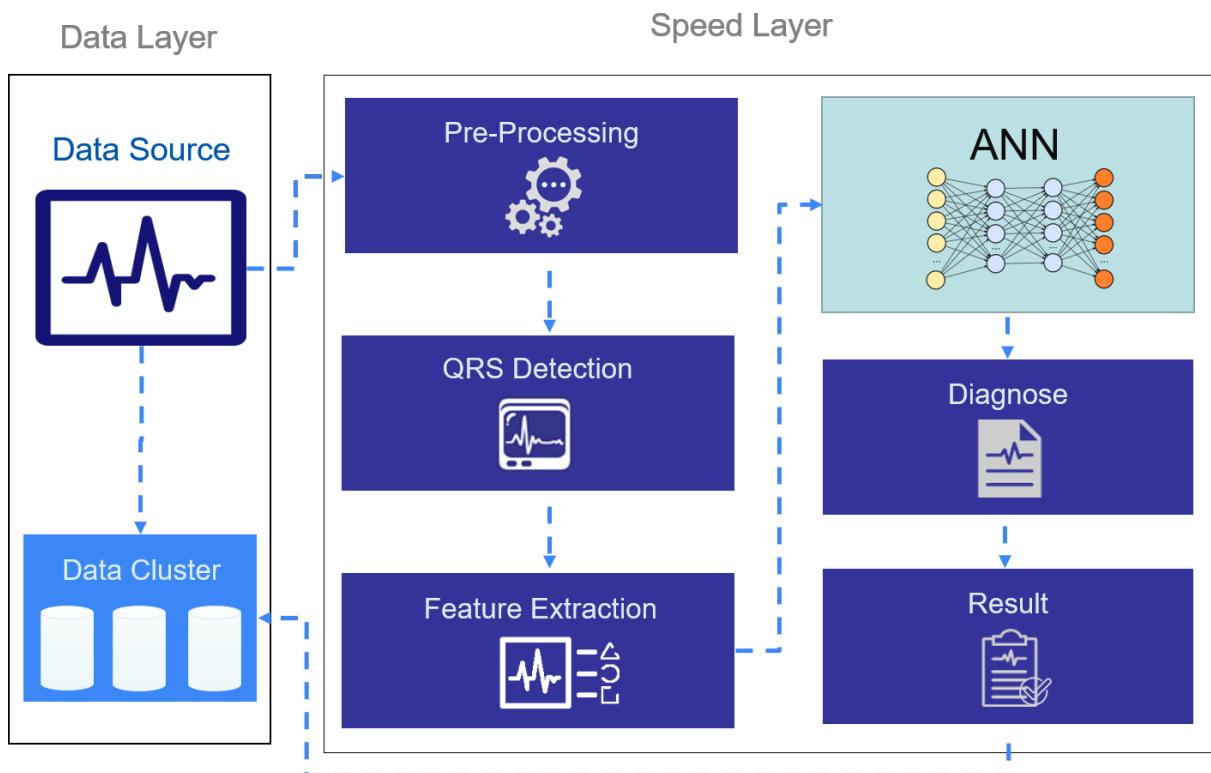


Figure. 5.9: Using the deep learning model in real-time environment.

The NI USB-6259 is used for the ECG and BCG signal acquisition. The sample rate of 360 Hz has been used for ECG signal as the deep learning model is trained on 360 Hz frequency data. The PPG and MI sensors data is collected via serial COM ports.

5.5.2 Speed Layer

The speed layer is the heart of the architecture where all the data is processed and classified by the CNN model. The ECG sensor data is first cleaned and the features are extracted from it. To extract the ECG signal features, the algorithm is used that is defined in Section 3.8. Once the R peaks are detected, the single ECG signal corresponding to that R peak consisting of P, Q, R, S and T peaks is extracted one by one, which is then passed to the deep learning model for the arrhythmia classification. The other vital signs are also calculated such as heart rate and temperature. The processed signals and the vital signs are also stored in the InfluxDB. The detail view of speed layer is shown in Figure 5.9.

5.5.3 Serving Layer

The serving layer allows the sensor signals and vital signs to be visualized. Grafana and android tablets are used for the visualization. Grafana retrieves the desired data from InfluxDB and displays it on the dashboard. Grafana provides many other options as well such as the refresh time. It specifies after how much time query for the new data.

The Grafana dashboards for ECG, MI and PPG sensors can be seen in Figure 5.10, 5.11 and 5.12 respectively.



Figure. 5.10: Real-time ECG signal visualization.

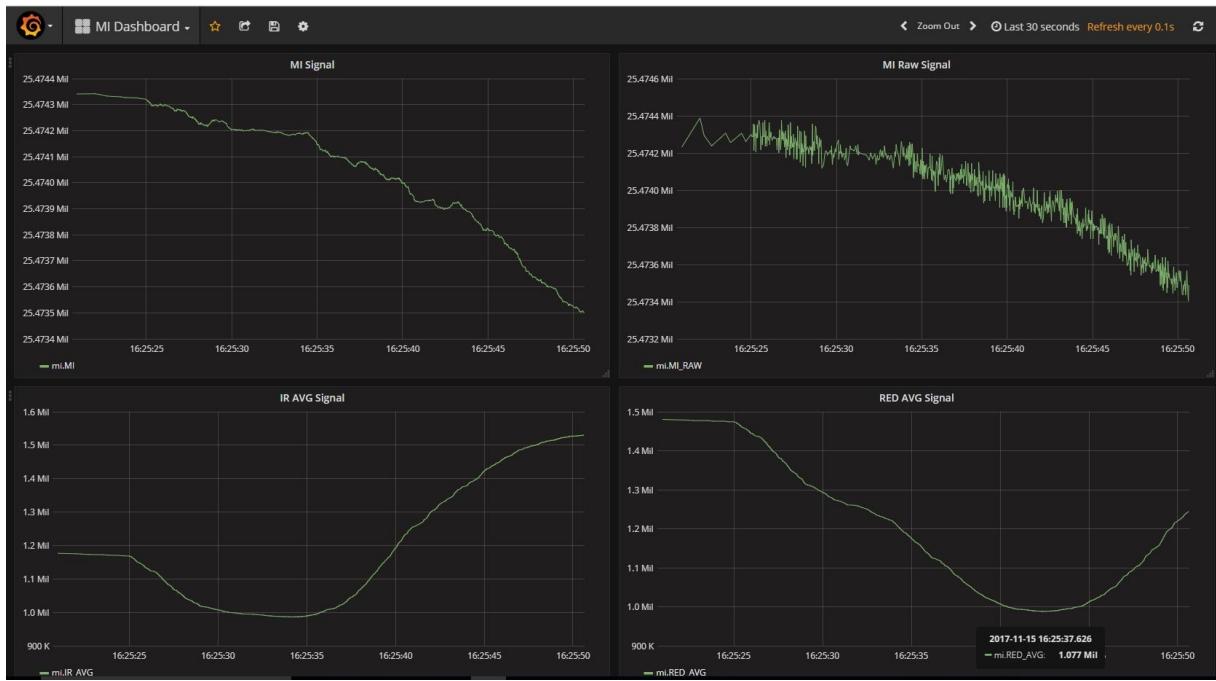


Figure. 5.11: Real-time MI signal visualization.



Figure. 5.12: Real-time PPG signal visualization.

Android Application

An android application has been written so the sensors data can be visualized directly on your mobile devices as well. The advantage of the Android application is that it is very lightweight and does not require any dependency to visualize the data. The screenshot of an Android application can be seen in Figure 5.13.

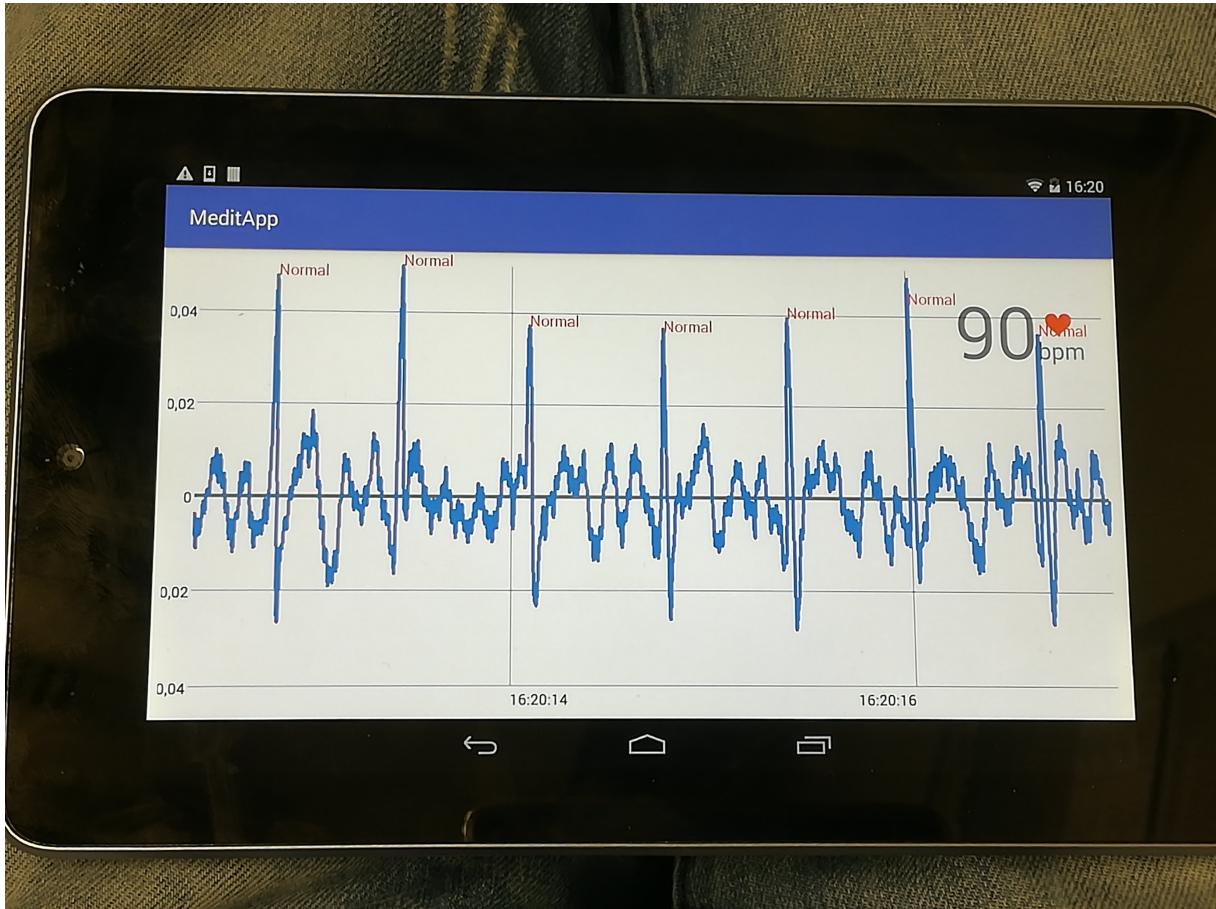
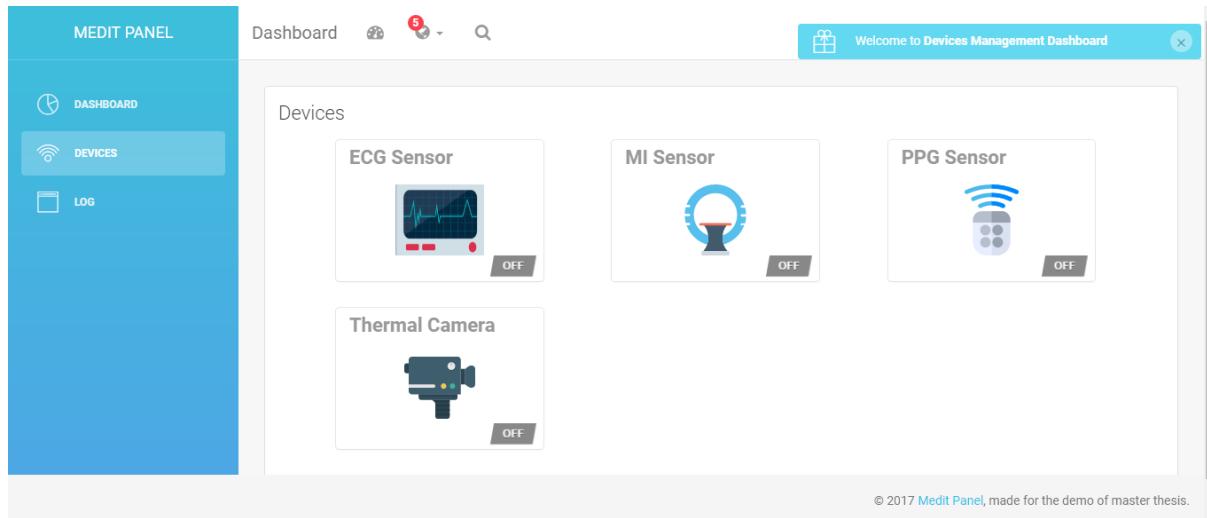


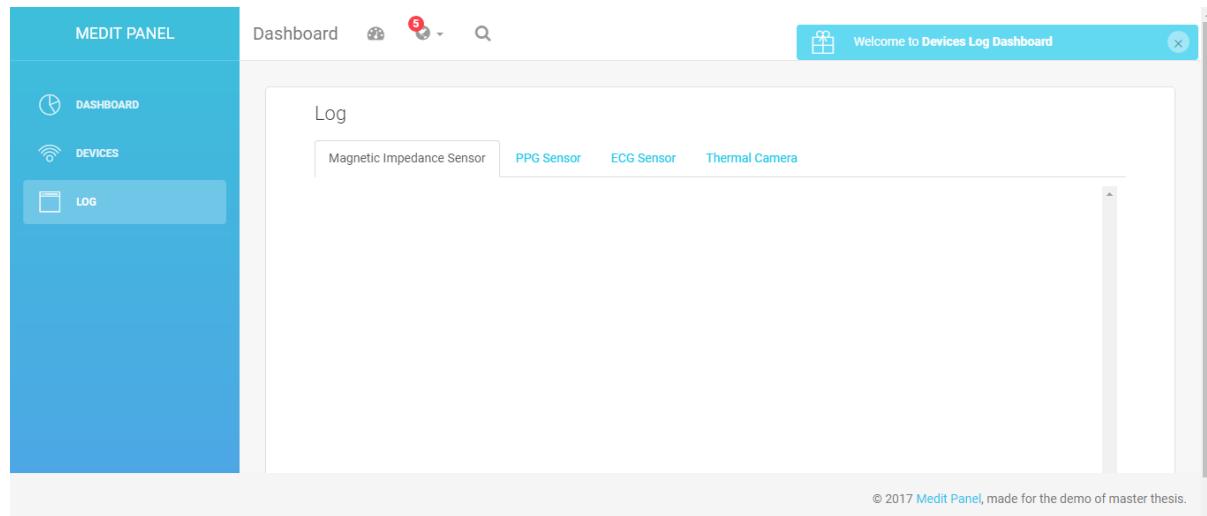
Figure. 5.13: Real-time ECG signal visualization on android application.

5.6 Device Management Interface

Starting the system via console is not a very convenient way to interact with the devices. Therefore, a simple web view interface has also been made to operate the sensor devices. The devices can be started, stopped and in case of error, the logs can be viewed directly in the same interface. The designed interface is shown in Figure 5.14.



(a)



(b)

Figure. 5.14: (a) The device management view to interact with devices; and, (b) the log view to know the state of each device.

6 Conclusion

In conclusion, a software system has been developed, that can measure various vital parameters of the pilot, including the accurate detection of QRS complex, and can detect cardiac conditions using CNN in real-time. The CNN model achieved an accuracy of 99.2%, whereas, in real-time, the model has an accuracy of 88%. The CNN model has been trained to classify four different types of arrhythmia. Moreover, our system can classify ECG signals in a real-time environment.

Main results and achievements of the work reported in this thesis can be summarized in the following points

In chapter 3, a detailed view of all the non-contact sensors has been explained, including their data protocol. A number of techniques, including digital filters and Wavelet transform, have been applied in order to process raw sensors signals, and remove noise and artifacts from them. An algorithm has been implemented based on Wavelet transform to extract the ECG features and important waves such P, QRS complex and T waves. The MIT-BIH arrhythmia dataset has been used for the development of the algorithm. The wavelet transform method has shown a high accuracy in the detection of QRS complexes and other waves, which is an essential step for ECG arrhythmias detection, classification, and extraction of vital parameters. The algorithm also evaluated on the chair's ECG signal, which gives an accurate location of all the waves in ECG signal.

In a sequel, a deep learning model has been proposed to identify the various cardiac conditions in chapter 4. CNN has been used for training the model, as it reduces the burden of extracting features from the signal, whereas, in traditional machine learning approaches, the programmers have to define which features should be considered while training the model. It mostly relies on the understanding of the programmers. Keras library has been used for training the CNN model, as it makes it very easy to define the layers for CNN model. It is very convenient to use, as it allows to choose a different type of backend for training the CNN such as Tensorflow or Theano. The arrhythmia dataset is extracted from the MIT-BIH dataset and used for the training of the model. The detail view of the model accuracy and its results have been presented, how many records are classified wrong and which records were classified as the other.

In chapter 5, different visualization tools and techniques have been described to visualize the vital parameters. Moreover, the system data architecture, which is based on big data lambda architecture, has been presented. Multiple sensors work together and gather various signals from the pilot and transferred them to the speed layer via data pipeline. The speed layer is the heart of the system architecture, where it processed and cleaned the signal. This layer is also responsible for extracting the features from the ECG signal. Once the signal is cleaned and processed, the CNN model classifies the ECG signal in real-time. The sensors signals, calculated parameters, and ECG signal classification is visualized on

Grafana, which is used for analytics and monitoring. It is one of the best tools for time series analytics and visualization. The sensors signal and vital parameters are also stored in the InfluxDB time series database.

A dashboard has been set up where the type of ECG signal classified by CNN model, sensors signals and vital parameters of the pilot can be monitored.

6.1 Future Work

There is still a potential to improve the overall software system. As mentioned in chapter 5, because of lack of resources, the big data tools have not used for the implementation of the system. But if the resources are available and the system is developed based on the big data tools such as Apache Kafka for handling the data from all the sensors, and Apache Spark to process all the sensors data in real-time, it will drastically improve the overall performance of the system. Big data can be organized in a much better way. The separation of the layers with their own responsibility makes the system much more efficient.

For now, the CNN model can detect four different types of arrhythmias, but the CNN model can be trained and extended for other cardiac conditions as well.

Another area for the improvement is to make the CNN model mode accurate in a real-time environment. In this thesis, the arrhythmia dataset has been used from the other source, but if the arrhythmia data is also collected from the chair, and then train the model using that data, will give much higher accuracy in the identification of the arrhythmia. On another note, the sensors are not much stable as they are very sensitive to use. For example, if MI sensor uses for quite a long time, it heats up and sometimes even provide wrong values. PPG sensor sometimes does not work for some different skin. One possible reason could be the temperature of the body. If the temperature is low, then it does not provide an accurate result. For the ECG measurement, if clothes are thick, it does not provide a correct signal, or the received signal has a lot of noise. There is still room for improvement in the sensors.

References

- [Add17] ADDISON, P.S.: *The Illustrated Wavelet Transform Handbook: Introductory Theory and Applications in Science, Engineering, Medicine and Finance, Second Edition.* CRC Press, 2017 <https://books.google.de/books?id=wBoNDgAAQBAJ>. – ISBN 9781482251333
- [AL07] ALEKSANDROWICZ, Adrian ; LEONHARDT, Steffen: Wireless and Non-contact ECG Measurement System—the Äachen SmartChair". In: *Acta Polytechnica* 47 (2007), Nr. 4-5
- [bil17] BILAGI: arrhythmia. <http://bilagi.org/blog/2016/08/24/chapter-9-ventricular-conduction-abnormalities/>. Version: 2017. – [Online; accessed 13-December-2017]
- [Bis06] BISHOP, C.M.: *Pattern Recognition and Machine Learning.* Springer, 2006 (Information Science and Statistics). <https://books.google.de/books?id=kTNQgAACAAJ>. – ISBN 9780387310732
- [Bou17] BOUCHARD, B.: *Smart Technologies in Healthcare.* CRC Press, 2017. – 55–56 S. <https://books.google.de/books?id=VgsqDwAAQBAJ>. – ISBN 9781498722018
- [Bro17] BROWNLEE, Jason: Supervised and Unsupervised Machine Learning Algorithms. <https://machinelearningmastery.com/supervised-and-unsupervised-machine-learning-algorithms/>. Version: 2017. – [Online; accessed 31-December-2017]
- [BSK16] B. SUNIL KUMAR, P. Suman R. J. Mohan Kumar2 K. J. Mohan Kumar2: Unified Big Data Lambda Architecture with Hadoop / Flume / Spark SQL, Streaming / Scala / Cassandra. 4 (2016), June. – ISSN 2320–9801
- [cab17] CABLESANDSENSORS: 12-Lead ECG Placement Guide with Illustrations. <https://www.cablesandsensors.com/pages/12-lead-ecg-placement-guide-with-illustrations>. Version: 2017. – [Online; accessed 31-December-2017]
- [CKL⁺06] CHIOU, Jin-Chern ; KO, Li-Wei ; LIN, Chin-Teng ; HONG, Chao-Ting ; JUNG, Tzzy-Ping ; LIANG, Sheng-Fu ; JENG, Jong-Liang: Using novel MEMS EEG sensors in detecting drowsiness application. In: *2006 IEEE Biomedical Circuits and Systems Conference*, 2006. – ISSN 2163–4025, S. 33–36
- [Com17a] COMMONS, Wikimedia: *File:Brantigan 1963 1-53.png* — Wikimedia Commons

- mons, the free media repository. https://commons.wikimedia.org/w/index.php?title=File:Brantigan_1963_1-53.png&oldid=229447194. Version: 2017. – [Online; accessed 8-November-2017]
- [Com17b] COMMONS, Wikimedia: *File:Limb leads.svg* — *Wikimedia Commons, the free media repository*. https://commons.wikimedia.org/w/index.php?title=File:Limb_leads.svg&oldid=262129086. Version: 2017. – [Online; accessed 8-November-2017]
- [Com17c] COMMONS, Wikimedia: *File:SinusRhythmLabels.svg* — *Wikimedia Commons, the free media repository*. <https://commons.wikimedia.org/w/index.php?title=File:SinusRhythmLabels.svg&oldid=260036882>. Version: 2017. – [Online; accessed 8-November-2017]
- [con17a] CONTRIBUTORS, Wikipedia: *Artificial neural network* — *Wikipedia, The Free Encyclopedia*. https://en.wikipedia.org/w/index.php?title=Artificial_neural_network&oldid=817006542. Version: 2017. – [Online; accessed 27-December-2017]
- [con17b] CONTRIBUTORS, Wikipedia: *Premature ventricular contraction* — *Wikipedia, The Free Encyclopedia*. https://en.wikipedia.org/w/index.php?title=Premature_ventricular_contraction&oldid=810563186. Version: 2017. – [Online; accessed 23-December-2017]
- [CSCB90] COAST, D. A. ; STERN, R. M. ; CANO, G. G. ; BRILLER, S. A.: An approach to cardiac arrhythmia analysis using hidden Markov models. In: *IEEE Transactions on Biomedical Engineering* 37 (1990), Sept, Nr. 9, S. 826–836. <http://dx.doi.org/10.1109/10.58593>. – DOI 10.1109/10.58593. – ISSN 0018-9294
- [Dow00] DOWDALL, Nigel: "Is there a doctor on the aircraft?" Top 10 in-flight medical emergencies. In: *BMJ* (2000), Nov, S. 1336–1337
- [E10] E, V.M.: *Cardiac Fibrillation-defibrillation: Clinical And Engineering Aspects*. World Scientific Publishing Company, 2010 (Series On Bioengineering And Biomedical Engineering). – 131–132 S. https://books.google.de/books?id=_L7FCgAAQBAJ. – ISBN 9789814465793
- [F⁺03] FORCE, Aerospace Medical Association Medical Guidelines T. u. a.: Medical guidelines for airline travel. In: *Aviation, space, and environmental medicine* 74 (2003), Nr. 5 Suppl, S. A1
- [Gre] GREG MORAN: *Chest Leads*. – [Online; accessed 8-November-2017]
- [Hau17] HAUSENBLAS, Michael: *Lambda Architecture*. <https://mapr.com/>

- developercentral/lambda-architecture/. Version: 2017. – [Online; accessed 31-December-2017]
- [IO17] ISIN, Ali ; OZDALILI, Selen: Cardiac arrhythmia detection using deep learning. In: *Procedia Computer Science* 120 (2017), 268 - 275. <http://dx.doi.org/https://doi.org/10.1016/j.procs.2017.11.238>. – DOI <https://doi.org/10.1016/j.procs.2017.11.238>. – ISSN 1877-0509. – 9th International Conference on Theory and Application of Soft Computing, Computing with Words and Perception, {ICSCCW} 2017, 22-23 August 2017, Budapest, Hungary
- [Iye17] IYER, Sulakshana: *Why is data visualization important.* <https://www.quora.com/Why-is-data-visualization-important>. Version: 2017. – [Online; accessed 29-December-2017]
- [JPM⁺16] JUN, T. J. ; PARK, H. J. ; MINH, N. H. ; KIM, D. ; KIM, Y. H.: Premature Ventricular Contraction Beat Detection with Deep Neural Networks. In: *2016 15th IEEE International Conference on Machine Learning and Applications (ICMLA)*, 2016, S. 859–864
- [JYK05] JEONG, Gu-Young ; YU, Kee-Ho ; KIM, Nam-Gyun: Continuous blood pressure monitoring using pulse wave transit time. In: *measurement* 4 (2005), Nr. 7
- [KLP04] KIM, Ko K. ; LIM, Yong K. ; PARK, Kwang S.: The electrically noncontacting ECG measurement on the toilet seat using the capacitively-coupled insulated electrodes. In: *Engineering in Medicine and Biology Society, 2004. IEMBS'04. 26th Annual International Conference of the IEEE* Bd. 1 IEEE, 2004, S. 2375–2378
- [KMM⁺15] KIRAN, M. ; MURPHY, P. ; MONGA, I. ; DUGAN, J. ; BAVEJA, S. S.: Lambda architecture for cost-effective batch and speed big data processing. In: *2015 IEEE International Conference on Big Data (Big Data)*, 2015, S. 2785–2792
- [Kre17] KREPS, Jay: *Questioning the Lambda Architecture.* <https://www.oreilly.com/ideas/questioning-the-lambda-architecture>. Version: 2017. – [Online; accessed 31-December-2017]
- [LD] LIMAYE, Mr H. ; DESHMUKH, Mrs V.: ECG Noise Sources and Various Noise Removal Techniques: A Survey.
- [LKP04] LIM, Yong K. ; KIM, Ko K. ; PARK, Kwang S.: The ECG measurement in the bathtub using the insulated electrodes. In: *Engineering in Medicine and Biology Society, 2004. IEMBS'04. 26th Annual International Conference of the IEEE* Bd. 1 IEEE, 2004, S. 2383–2385

- [LU18] LEENA UKKONEN, Lauri S.: *Contactless health-care sensing*. <https://www.nature.com/articles/d41586-017-07434-y>. Version: 2018. – [Online; accessed 7-January-2018]
- [mau17] MAUVILA: *ECG artifacts*. http://www.mauvila.com/ECG/ecg_artifact.htm. Version: 2017. – [Online; accessed 12-November-2017]
- [McD18] McDONALD, Carol: *Applying Machine Learning to Streaming IoT for Connected Medical Devices*. <https://mapr.com/blog/ml-iot-connected-medical-devices/>. Version: 2018. – [Online; accessed 7-January-2018]
- [med17] MEDICINENET: *arrhythmia*. https://www.medicinenet.com/arrhythmia_irregular_heartbeat/article.htm. Version: 2017. – [Online; accessed 13-December-2017]
- [MG13] MARKOVSKI, S. ; GUSEV, M.: *ICT Innovations 2012: Secure and Intelligent Systems*. Springer Berlin Heidelberg, 2013 (Advances in Intelligent Systems and Computing). <https://books.google.de/books?id=mWtHAAAQBAJ>. – ISBN 9783642371691
- [MH92] MALLAT, S. ; HWANG, W. L.: Singularity detection and processing with wavelets. In: *IEEE Transactions on Information Theory* 38 (1992), March, Nr. 2, S. 617–643. <http://dx.doi.org/10.1109/18.119727>. – DOI 10.1109/18.119727. – ISSN 0018–9448
- [ML08] MANTZIARI L, Kourtidou-Papadeli C Styliadis I. Styliadis C C. Styliadis C: Arrhythmias, Sudden Cardiac Death and incapacitation of pilots. In: *Hippokratia* Suppl 1 (2008), Aug, Nr. 12, S. 53–58
- [MPS⁺11] MITHUN, P. ; PANDEY, P. C. ; SEBASTIAN, T. ; MISHRA, P. ; PANDEY, V. K.: A wavelet based technique for suppression of EMG noise and motion artifact in ambulatory ECG. In: *2011 Annual International Conference of the IEEE Engineering in Medicine and Biology Society*, 2011. – ISSN 1094–687X, S. 7087–7090
- [MRF13] MAHDAVI, H ; ROSELL-FERRER, J: A magnetic induction measurement system for adult vital sign monitoring: evaluation of capacitive and inductive effects. In: *Journal of Physics: Conference Series* 434 (2013), Nr. 1, 012085. <http://stacks.iop.org/1742-6596/434/i=1/a=012085>
- [Neu98] NEUMAN, Michael R.: Biopotential amplifiers. In: *Medical instrumentation: application and design* (1998), S. 292–296
- [Pie91] PIETKA, Ewa: Feature extraction in computerized approach to the ecg

- analysis. In: *Pattern Recognition* 24 (1991), Nr. 2, 139 - 146. [http://dx.doi.org/https://doi.org/10.1016/0031-3203\(91\)90083-H](http://dx.doi.org/https://doi.org/10.1016/0031-3203(91)90083-H). – DOI [https://doi.org/10.1016/0031-3203\(91\)90083-H](https://doi.org/10.1016/0031-3203(91)90083-H). – ISSN 0031–3203
- [PRK16] POURBABAE, B. ; ROSHTKHARI, M. J. ; KHOASANI, K.: Feature learning with deep Convolutional Neural Networks for screening patients with paroxysmal atrial fibrillation. In: *2016 International Joint Conference on Neural Networks (IJCNN)*, 2016, S. 5057–5064
- [PT85] PAN, J. ; TOMPKINS, W. J.: A Real-Time QRS Detection Algorithm. In: *IEEE Transactions on Biomedical Engineering* BME-32 (1985), March, Nr. 3, S. 230–236. <http://dx.doi.org/10.1109/TBME.1985.325532>. – DOI [10.1109/TBME.1985.325532](https://doi.org/10.1109/TBME.1985.325532). – ISSN 0018–9294
- [PZZ10] PAN, T. ; ZHANG, L. ; ZHOU, S.: Detection of ECG characteristic points using Biorthogonal Spline Wavelet. In: *2010 3rd International Conference on Biomedical Engineering and Informatics* Bd. 2, 2010. – ISSN 1948–2914, S. 858–863
- [QDFM06] QIU, Yazhu ; DING, Xianfeng ; FENG, Jun ; MO, Zhiwen: [QRS complexes detection based on Mexican-hat wavelet]. In: *Sheng wu yi xue gong cheng xue za zhi = Journal of biomedical engineering = Shengwu yixue gongchengxue zazhi* 23 (2006), December, Nr. 6, 1347?1349. <http://europepmc.org/abstract/MED/17228741>. – ISSN 1001–5515
- [RSN97] RUHA, A. ; SALLINEN, S. ; NISSLIA, S.: A real-time microprocessor QRS detector system with a 1-ms timing accuracy for the measurement of ambulatory HRV. In: *IEEE Transactions on Biomedical Engineering* 44 (1997), March, Nr. 3, S. 159–167. <http://dx.doi.org/10.1109/10.554762>. – DOI [10.1109/10.554762](https://doi.org/10.1109/10.554762). – ISSN 0018–9294
- [Sch12] SCHILLING, C.: *Analysis of Atrial Electrograms*. KIT Scientific Publ., 2012 (Karlsruhe transactions on biomedical engineering). <https://books.google.de/books?id=rutA1A-ia4oC>. – ISBN 9783866448940
- [Sch17] SCHOOLINFO: *Electrical activity of Heart*. <http://schoolbag.info/biology/mcat/32.html>. Version: 2017. – [Online; accessed 12-November-2017]
- [SDC07] SULLIVAN, Thomas J. ; DEISS, Stephen R. ; CAUWENBERGHS, Gert: A low-noise, non-contact EEG/ECG sensor. In: *Biomedical Circuits and Systems Conference, 2007. BIOCAS 2007. IEEE IEEE*, 2007, S. 154–157
- [SLL14] SHANG, Yu ; LEI, Shasha ; LIU, Bin: QRS Characteristic Waveform Extraction

- Based On Biorthogonal B-Spline Wavelet. In: *International Journal of Control and Automation* 7 (2014), Nr. 1, S. 95–106
- [SN96] STRANG, G. ; NGUYEN, T.: *Wavelets and Filter Banks*. Wellesley-Cambridge Press, 1996 https://books.google.de/books?id=Z76N_Ab5pp8C. – ISBN 9780961408879
- [SSLO07] SHEN, Tsu-Wang ; SHEN, Hsiao-Ping ; LIN, Ching-Heng ; OU, Yi-Ling: Detection and prediction of sudden cardiac death (SCD) for personal healthcare. In: *Engineering in Medicine and Biology Society, 2007. EMBS 2007. 29th Annual International Conference of the IEEE IEEE*, 2007, S. 2575–2578
- [sum17] SUMDU: arrhythmia. https://elearning.sumdu.edu.ua/free_content/lectured:a3664646bdd2fb5e12ac4d9fc4d3dff83793be56/20150113065426//64025/index.html. Version: 2017. – [Online; accessed 13-December-2017]
- [TWT84] THAKOR, N. V. ; WEBSTER, J. G. ; TOMPKINS, W. J.: Estimation of QRS Complex Power Spectra for Design of a QRS Filter. In: *IEEE Transactions on Biomedical Engineering* BME-31 (1984), Nov, Nr. 11, S. 702–706. <http://dx.doi.org/10.1109/TBME.1984.325393>. – DOI 10.1109/TBME.1984.325393. – ISSN 0018–9294
- [ujj17] UJJWALKARN: A Quick Introduction to Neural Networks. <https://ujjwalkarn.me/2016/08/09/quick-intro-neural-networks/>. Version: 2017. – [Online; accessed 27-December-2017]
- [Wil05] WILKINS, L.W.: *ECG Interpretation Made Incredibly Easy*. Lippincott Williams & Wilkins, 2005 (Incredibly Easy! Series). <https://books.google.de/books?id=mGiTP7zCHUAC>. – ISBN 9781582553559
- [WLH01] WANG, Chaowen ; LU, Yangsheng ; HUANG, Yuxi: Using biorthogonal wavelet filter bank for ECG detection and reconstruction. In: *Beijing Biomedical Engineering* 20 (2001), Nr. 1, S. 25–28
- [WZ08] WU, Kin-fai ; ZHANG, Yuan-ting: Contactless and continuous monitoring of heart electric activities through clothes on a sleeping bed. In: *Information Technology and Applications in Biomedicine, 2008. ITAB 2008. International Conference on IEEE*, 2008, S. 282–285
- [XHT92] XUE, Q. ; HU, Y. H. ; TOMPKINS, W. J.: Neural-network-based adaptive matched filtering for QRS detection. In: *IEEE Transactions on Biomedical Engineering* 39 (1992), April, Nr. 4, S. 317–329. <http://dx.doi.org/10.1109/10.126604>. – DOI 10.1109/10.126604. – ISSN 0018–9294