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: 5. Januar 2018

Acknowledgments

Lorem ipsum dolor sit amet, consetetur sadipscing elitr, sed diam nonumy eirmod tempor invidunt ut labore et dolore magna aliquyam erat, sed diam voluptua. At vero eos et accusam et justo duo dolores et ea rebum. Stet clita kasd gubergren, no sea takimata sanctus est Lorem ipsum dolor sit amet. Lorem ipsum dolor sit amet, consetetur sadipscing elitr, sed diam nonumy eirmod tempor invidunt ut labore et dolore magna aliquyam erat, sed diam voluptua. At vero eos et accusam et justo duo dolores et ea rebum. Stet clita kasd gubergren, no sea takimata sanctus est Lorem ipsum dolor sit amet. Lorem ipsum dolor sit amet, consetetur sadipscing elitr, sed diam nonumy eirmod tempor invidunt ut labore et dolore magna aliquyam erat, sed diam voluptua. At vero eos et accusam et justo duo dolores et ea rebum. Stet clita kasd gubergren, no sea takimata sanctus est Lorem ipsum dolor sit amet.

Duis autem vel eum iriure dolor in hendrerit in vulputate velit esse molestie consequat, vel illum dolore eu feugiat nulla facilisis at vero eros et accumsan et iusto odio dignissim qui blandit praesent luptatum zzril delenit augue duis dolore te feugait nulla facilisi. Lorem ipsum dolor sit amet, consectetuer adipiscing elit, sed diam nonummy nibh euismod tincidunt ut laoreet dolore magna aliquam erat volutpat.

Ut wisi enim ad minim veniam, quis nostrud exerci tation ullamcorper suscipit lobortis nisl ut aliquip ex ea commodo consequat. Duis autem vel eum iriure dolor in hendrerit in vulputate velit esse molestie consequat, vel illum dolore eu feugiat nulla facilisis at vero eros et accumsan et iusto odio dignissim qui blandit praesent luptatum zzril delenit augue duis dolore te feugait nulla facilisi.

Eidesstattliche Versicherung

Name, Vorname | Matrikelnummer (freiwillige Angabe)

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

selbständig 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

Lorem ipsum dolor sit amet, consetetur sadipscing elitr, sed diam nonumy eirmod tempor invidunt ut labore et dolore magna aliquyam erat, sed diam voluptua. At vero eos et accusam et justo duo dolores et ea rebum. Stet clita kasd gubergren, no sea takimata sanctus est Lorem ipsum dolor sit amet. Lorem ipsum dolor sit amet, consetetur sadipscing elitr, sed diam nonumy eirmod tempor invidunt ut labore et dolore magna aliquyam erat, sed diam voluptua. At vero eos et accusam et justo duo dolores et ea rebum. Stet clita kasd gubergren, no sea takimata sanctus est Lorem ipsum dolor sit amet. Lorem ipsum dolor sit amet, consetetur sadipscing elitr, sed diam nonumy eirmod tempor invidunt ut labore et dolore magna aliquyam erat, sed diam voluptua. At vero eos et accusam et justo duo dolores et ea rebum. Stet clita kasd gubergren, no sea takimata sanctus est Lorem ipsum dolor sit amet.

Duis autem vel eum iriure dolor in hendrerit in vulputate velit esse molestie consequat, vel illum dolore eu feugiat nulla facilisis at vero eros et accumsan et iusto odio dignissim qui blandit praesent luptatum zzril delenit augue duis dolore te feugait nulla facilisi. Lorem ipsum dolor sit amet, consectetuer adipiscing elit, sed diam nonummy nibh euismod tincidunt ut laoreet dolore magna aliquam erat volutpat.

Ut wisi enim ad minim veniam, quis nostrud exerci tation ullamcorper suscipit lobortis nisl ut aliquip ex ea commodo consequat. Duis autem vel eum iriure dolor in hendrerit in vulputate velit esse molestie consequat, vel illum dolore eu feugiat nulla facilisis at vero eros et accumsan et iusto odio dignissim qui blandit praesent luptatum zzril delenit augue duis dolore te feugait nulla facilisi.

Table of Contents

List of Figures

List of Tables

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

Health care is one of the hottest research areas in the current era. Monitoring of vital signs parameter like respiration, ECG, EEG, temperature, heart rate, etc,. getting more and more important in every field of life. The most focus is towards the wearable and wireless sensor as they do the minimal obstruction in day to day life.

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

Contact-less sensors are the next big thing in the health care. They are designed in such a way that they can disappear in the daily surroundings without any disruption. Different techniques has been used to integrate contact-less sensors into clothes, chairs, smart watches, smart phones, shoes, beds, etc ot measure different vital signs.

As mentioned earlier, most of the work done in the health care technologies have uses the sensors which are required to be placed on the body, therefore, a trends need to be shift towards contact-less sensors as they are much more convenient to use. Moreover, not much work has been carried out to use the deep learning technologies along with contact-less sensor to identify the cardiac arrhythmias in real-time environment.

The aim of this thesis, is to build a system that provides a health care solution which would be able to track the pilot's vital signs including heart rate, temperature, ECG, etc, as well as can detect the cardiac arrhythmias in real-time. The early identification of disease can help to provide quick treatment to the pilot.

Multiple contact-less sensors are used in the development of the system to measure the various vital signs of pilots. These sensors include:

1. Capactive ECG sensor
2. Photoplethysmogram sensor
3. Magnetic impedance sensor
4. Ballistocardiogram sensor

5. Thermal camera

These sensors will be installed in the pilot seat so they can continuously track the health status of pilot during a course of flight, and in case of any irregularity, immediate actions can be taken for the safety of the pilot and of course for the passengers as well.

A deep learning model has been trained using convolutional neural network. Various cardiac arrhythmias dataset were taken from already existed dataset to train the model. 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 and vital signs of the passengers. A cloud has been setup to store all the sensors data and vital signs so, the data can be accessed from anywhere in the world. The same data later can also be used for different purpose such as, for the training of deep learning model.

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 ventricles 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. Synoatrial (SA) node
2. The Atrioventricular (AV) node
3. Atrioventricular (AV) bundle or bundle of His
4. Right and left bndle 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 spread 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.

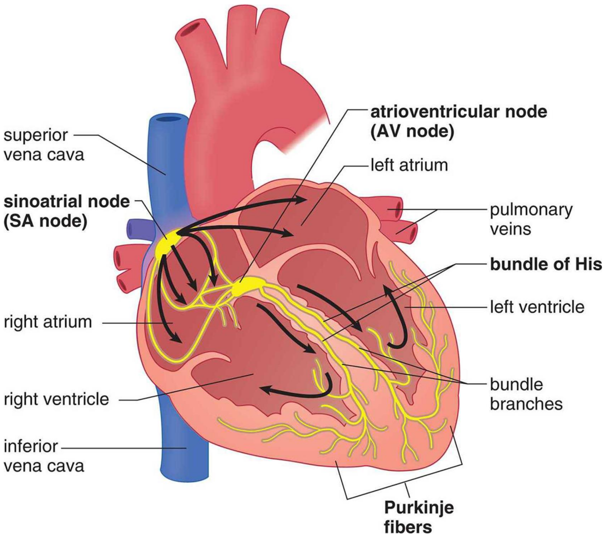


Figure. 2.1: The electrical activity of heart[Sch17].

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

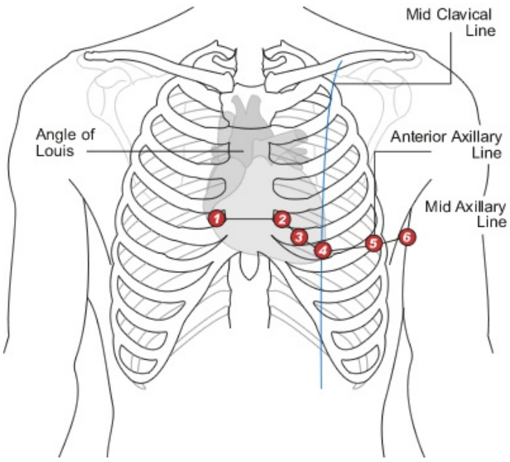


Figure. 2.2: The leads position on chest[Gre].

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.

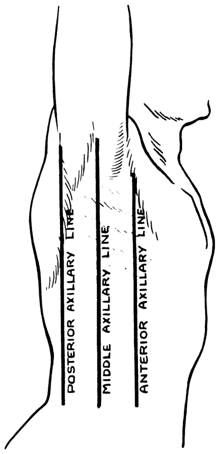


Figure. 2.3: The axillary lines on the right side of chest[Com17a].

Limb Electrodes

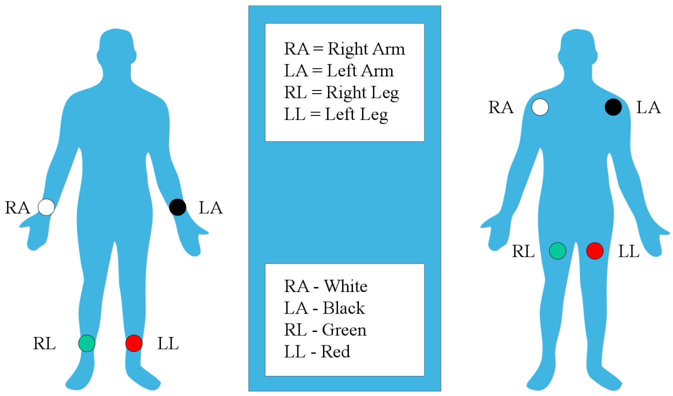


Figure. 2.4: The possible position of limb leads[Com17b].

The 4 limb electrodes are denoted as RA, LA, LL, RL and there 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

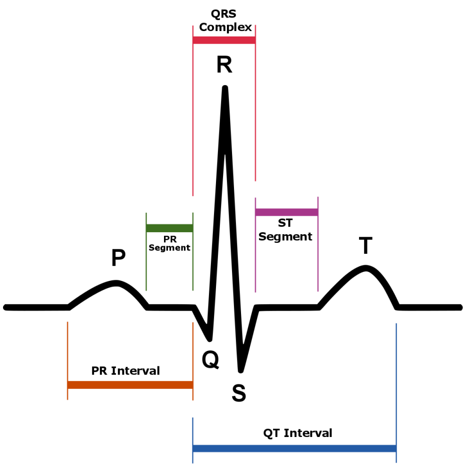


Figure. 2.5: The ECG signal[Com17c].

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 create a pulse. The Q and S waves are relatively very 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 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.

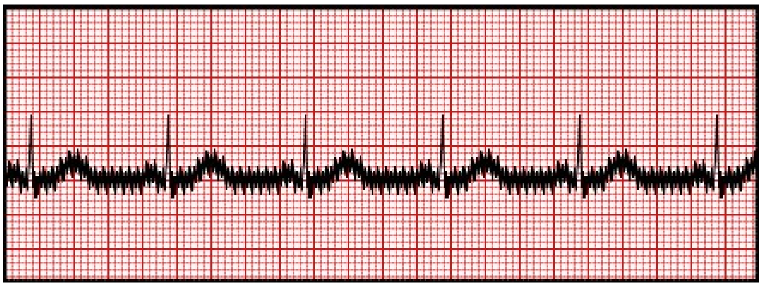


Figure. 2.6: AC Interference of 50Hz[mau17].

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

Arrhytmia 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”. 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.

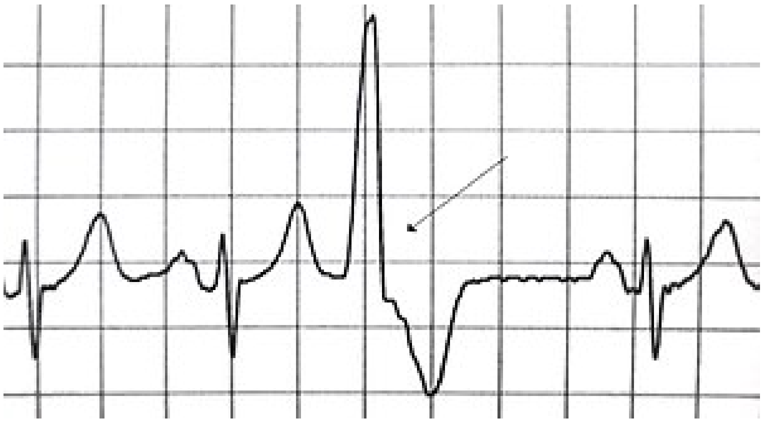


Figure. 2.9: Premature Ventricular Contraction[con17b].

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 palpation, 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.

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.

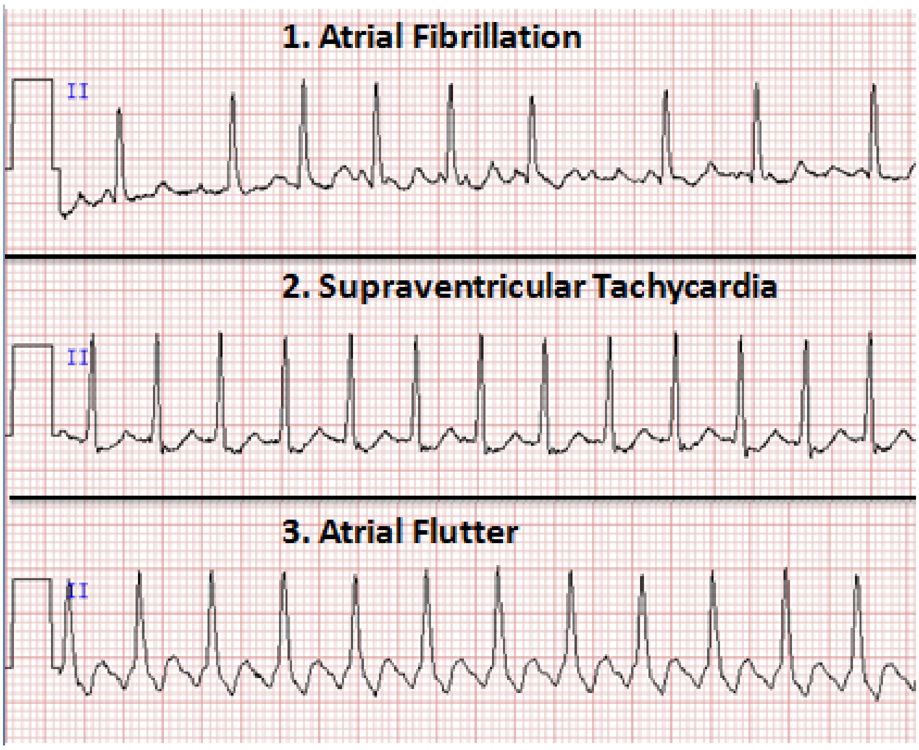


Figure. 2.10: Atrial Fibrillation, Atrial Flutter and Tachycardia[sum17].

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 ECG Signal and Data Processing

3.1 Devices

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 health care system which not only can be used in hospitals but also can be utilized out of the hospital when the doctor is unreachable regardless of the patient's location. Doctors will still be able to monitor his patient condition and according to the patient situation they can instruct the device, that is, attached to the patient, to take appropriate actions. One example can be thought of as a person running somewhere and during that he/she feels some heart pain. Sensors assess the patient's condition and immediately send some notification to the doctor. After looking at the conditions, he sends back a response to the devices, which then acts according to the instruction such as, injects some medicine into the patient body. It can also be used to keep track of the patient location so, in the case of emergency, an ambulance can be instructed to go there. Many of the sensors can be installed in the patient's surrounding, whereas, several sensors can be wearable. These sensors can monitor body temperature, respiration, heart rate, blood pressure, ECG, EEG, etc. Along with sensors, it might be possible that there are several actuators attached to the patient body which is activated by certain events such as the rise of sugar level in blood.

Multiple contact-less sensor devices are used to implement a system for the thesis, which collects data of the user and process that in real-time. The following devices are used in the implementation:

3.1.1 Magnetic Impedance Sensor

The magnetic impedance (MI) sensor measures the small changes in electrical resistance of the chest or different regions of the body. It uses special electrodes which emit very low voltage electric current into the body. As the voltage level is very low, therefore, it does not interfere with the heart's electrical system. The MI sensor measures the resistance to the flow of current as it passes through the body via blood, as blood is a good conductor.

During systole, as the blood volume increases, impedance decreases. Similarly, during diastole, the blood volume decreases and the impedance increases.

The MI sensor provides a data packet of 42 bytes, which splits into the attributes shown in table 3.1. The byte identifier of the sensor can be seen in the table 3.2.

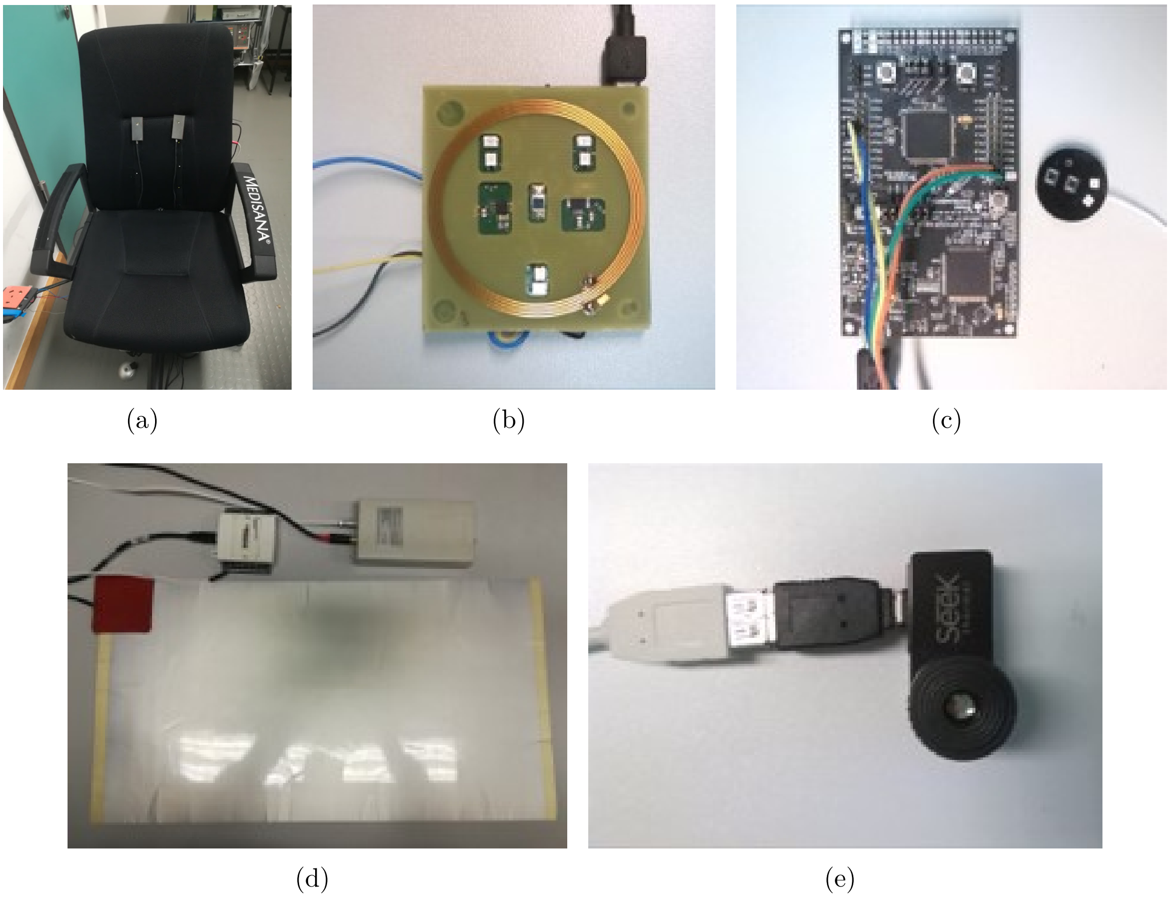


Figure. 3.1: A set of sensor devices: (a) an ECG sensor; (b) a MI sensor; (c) a PPG sensor; (a) a BCG sensor; and, (b) a thermal camera.

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

There is an exception in the data packet when it contains 0x8101 or 0x8102. In this case, the size of the data packet may vary according to the number of count of the corresponding bytes. Moreover, the data packet should be examined and if it contains the corresponding bytes, the data packet should be modified and the bytes 0x8101 should be replaced with 0x81 and 0x8102 should be replaced with 0x82. This modification has been done in order to differentiate it with the header and data packet identifier as they have the same value.

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. A PPG sensor uses a light source to illuminate the blood and a photo-detector to measures the variations in the light intensity associated with changes in the blood volume. The decrease in light intensity indicates the increase in blood volume and increase in light intensity indicates the decrease in blood volume.

The sensor provides the PPG signal with 4 different channels, a temperature, which is measured in Celsius and accelerometer coordinates. The size of the data changes according to the attributes which can be seen in the table 3.3. The frequency of the data packets changes according to the attributes. The temperature value is sent every one second, whereas, the frequency rate of PPG channels is 100 samples per second. Similarly, for the accelerometer coordinates, the data rate is 50 samples per second.

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 well, which is described in section 2.4. Therefore, non-contact capacitive electrodes have been used to collect the ECG signals of the person. Unlike traditional electrodes, which rely on galvanic contact, the capacitive electrodes are insulated from skin using a dielectric material, such as, air gap, clothes, etc [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 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 measures by the elec-

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

tromechanical film (EMFi) sensor which converts the mechanical energy into the electrical signal and vice versa. 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 measures the temperature of the person. A thermal seek camera is used for the implementation which captures the thermo temperature images, from which then the temperature is calculated.

3.2 ECG Signal Processing

QRS complex detection is the basis for processing ECG signal. Regardless of what application is required, the accurate detection of QRS complex is a pre-requisite 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 examination of the signal can be performed such as heart rate, arrhythmias, classification of ECG signal, ST segment etc. Moreover, P and T waves can also be extracted.

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 [PZZ10a].

1. Filter Method

2. Artificial Intelligence Method

3. Wavelet transform Method

3.2.1 Filter Method

The filter method uses 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 very 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 very popular in detecting the QRS complex. It is based on time-frequency analysis. It is very efficient and takes less to implement. Many people 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, but it sometimes didn’t detect the onset and offset of QRS complex accurately. Nevertheless, it is considered as simple, faster and easier to implement comparatively.

In the implementation of this thesis, the wavelet transform method has been used to extract the ECG signal features.

3.3 Wavelet Transform

Transformation is applied to signal in order to get further information about the signal which is not easily available in the raw signal. Most of the time, signals are generally represented in the time domain, but in many cases, the important information is hidden in the frequency domain of the signal. Fourier transform is a tool which allows viewing the frequency components of the signal. But the major drawback of this transformation is that a signal cannot be viewed in both time and frequency domain at the same time. Thus, it makes hard to distinguish which frequency components exist at any instance of time. Therefore, a tool was required which helps to view signal in both domains.

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 wavelets". It is defined as:

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

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 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 is defined as:

$$X_W(a,b) = \frac{1}{\sqrt{|a|}}\int_{-\infty}^{\infty} h * \left(\frac{t-b}{a}\right)x(t)\mathrm{d}x \tag{3.2}$$

Biorthogonal Spline wavelet has been used for the detection of ECG signal characteristics

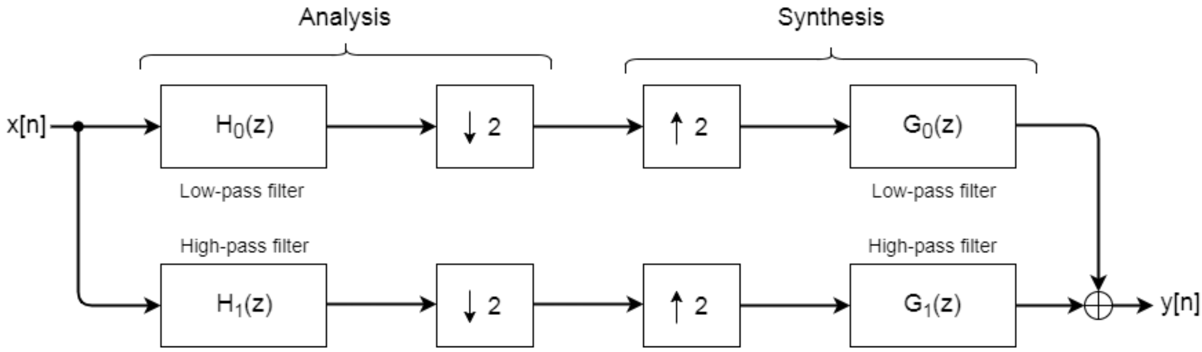


Figure. 3.2: Two-channel filter bank

[ZLSJ08] [PZZ10b] [MS14]. This approach is based on the modulus maxima of zero point to detect the singular point.

For multi-resolution decomposition of signals, a dyadic DWT (discrete wavelet transform) is used where all bandpass filters have different frequency resolution. This is done by first using low-pass and high-pass filters to split the signal into low and high-frequency components.

3.4 Biorthogonal Spline Wavelet Filter Construction

Let say, $H_0(z)$ and $H_1(z)$ are low-pass filters in the analysis filter bank (decomposition) and $G_0(z)$, $G_1(z)$ are high-pass filters in the synthesis filter bank (reconstruction) [WLH01], as can be seen 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 $Y[n]$. It is worth to note here that $Y[n]$ is the reconstructed signal.

The idea is to determine H_0, H_1, G_0 and G_1 such that, $Y[n]$ is just 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.

After passing the input signal from the channel 1, it will produce:

$$Y_0(z) = \frac{1}{2}G_0(z)[H_0(z)X(z) + H_0(-z)X(-z)] \tag{3.3}$$

Similarly, for the 2nd channel, it will produce:

$$Y_1(z) = \frac{1}{2}G_1(z)[H_1(z)X(z) + H_1(-z)X(-z)] \tag{3.4}$$

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

$$\begin{aligned} Y(z) &= Y_0(z) + Y_1(z) \\ &= \frac{1}{2}G_0(z)[H_0(z)X(z) + H_0(-z)X(-z)] + \frac{1}{2}G_1(z)[H_1(z)X(z) + H_1(-z)X(-z)] \end{aligned} \tag{3.5}$$

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

$$Y(z) = \frac{1}{2}[G_0(z)H_0(z) + G_1(z)H_1(z)]X(z) + \frac{1}{2}[G_0(z)H_0(-z) + G_1(z)H_1(-z)]X(-z) \tag{3.6}$$

The important thing to note here is that $X(-z)$ is the aliasing part, and $X(z)$ is the distortion part.

3.4.1 Design of Biorthogonal Spline Wavelet Filter

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

1. No aliasing:
$$G_0(z)H_0(-z) + G_1(z)H_1(-z)]X(-z) = 0 \tag{3.7}$$

2. No distortion:
$$G_0(z)H_0(z) + G_1(z)H_1(z) = mz^{-k} \tag{3.8}$$

where m is constant and k is a time delay.

In order to satisfy condition 1 i.e., to get rid of aliasing, one can do:

$$\begin{aligned} G_0(z) &= H_1(-z), \\ G_1(z) &= -H_0(-z) \end{aligned} \tag{3.9}$$

So now, we only need to find two filters values instead of four. Lets assume that,

$$P_0(z) = G_0(z)H_0(z) \tag{3.10}$$

From equation 3.9 and 3.10, we can deduce:

$$G_1(z)H_1(z) = -H_0(-z)G_0(-z) = -P_0(-z) \tag{3.11}$$

After getting these values, the condition 2 (no distortion) can be re-written as:

$$P_0(z) - P_0(-z) = mz^{-k} \tag{3.12}$$

In the above equation, only one filter value is required i.e., $P_0(z)$ (also called half-band filter). The perfect reconstruction conditions naturally imply that both analysis and the synthesis filters are biorthogonal to each other, i.e., a biorthogonal filter bank makes sure that synthesis filter bank is the inverse of analysis filter bank.

3.4.2 Steps for Designing FIR Filter Bank

The steps for designing FIR filter bank can be summarized as:

1. Design a low-pass filter for $P_0(z)$ which satisfy the equation 3.13. One option is to use Daubechies function to find the value for $P_0(z)$:

   $$P_0(z) = (1 + z^{-1})^{2p}Q(z) \tag{3.13}$$

   where p can be any integer and $Q(z)$ be a polynomial of degree $(2p - 2)$.

2. Factorize $P_0(z)$ to get the values for $G_0(z)$ and $H_0(z)$.

3. Find the filter coefficients for high-pass filters using the equations 3.9.

Lets assume that, $P = 2$ and and $Q(z)$ be a quadratic polynomial $(a + bz^{-1} + cz^{-2})$. Substituting these values in equation 3.13 will produce a polynomial of degree z:

$$P_0(z) = (1 + z^{-1})^4(a + bz^{-1} + cz^{-2}) \tag{3.14}$$

Substituting $a = c = -\frac{1}{16}, b = 4$, we get:

$$P_0(z) = \frac{(1+z^{-1})^4(-1+4z^{-1}+z^{-2})}{16} \tag{3.15}$$

Factorizing $P_0(z)$ to get $H_0(z)$ and $G_0(z)$. Lets say we get:

$$\begin{aligned} H_0(z) &= \frac{(1+z^{-1})^3}{4} \\ &= \frac{(1+3z^{-1}+3z^{-2}+z^{-3})}{4} \end{aligned} \tag{3.16}$$

and

$$\begin{aligned} G_0(z) &= \frac{(1+z^{-1})(-1+4z^{-1}+z^{-2})}{4} \\ &= \frac{(-1+3z^{-1}+3z^{-2}-z^{-3})}{4} \end{aligned} \tag{3.17}$$

Then by equation 3.13, we have:

$$\begin{aligned} H_1(z) = G_0(-z) \\ &= \frac{(-1-3z^{-1}+3z^{-2}+z^{-3})}{4} \end{aligned} \tag{3.18}$$

and

$$\begin{aligned} G_1(z) = -H_0(-z) \\ &= \frac{(-1+3z_3^{-1}z^{-2}+z^{-3})}{4} \end{aligned} \tag{3.19}$$

Therefore, the filter coefficients are:

$$
\begin{aligned}
h_0(0) &= \frac{1}{4} & h_0(1) &= \frac{3}{4} \\
h_0(2) &= \frac{3}{4} & h_0(3) &= \frac{1}{4} \\
h_1(0) &= \frac{-1}{4} & h_1(1) &= \frac{-3}{4} \\
h_1(2) &= \frac{3}{4} & h_1(3) &= \frac{1}{4} \\
g_0(0) &= \frac{-1}{4} & g_0(1) &= \frac{3}{4} \\
g_0(2) &= \frac{3}{4} & g_0(3) &= \frac{-1}{4} \\
g_1(0) &= \frac{-1}{4} & g_1(1) &= \frac{3}{4} \\
g_1(2) &= \frac{-3}{4} & g_1(3) &= \frac{1}{4}
\end{aligned}
\tag{3.20}
$$

3.5 Mallat's Algorithm

The binary wavelet transform or dyadic wavelet transform of a signal $f(n)$ can be calculated by using Mallat algorithm [MH92] as follows:

$$s_{2^j} f(n) = \sum h_k s_{2^{j-1}} f(n - 2^{j-1}k), \tag{3.21}$$

$$w_{2^j} f(n) = \sum g_k s_{2^{j-1}} f(n - 2^{j-1}k). \tag{3.22}$$

where, $s_{2^0} f(n)$ is the original signal to be processed. In our case, it is ECG signal. $w_{2^j} f(n)$ is the wavelet coefficient i.e., the dyadic wavelet transform of the signal and $s_{2^j} f(n)$ is the approximation coefficient for the scale j. h_k and g_k are the coefficients of a low-pass filter and high-pass filter respectively which are defined in the equation 3.20. The signal is decomposed into several frequency bands at certain scale j and then the frequency bands which have noises, are set to zero. And by using the synthesis filters, the de-noised signal can be reconstructed.

3.6 Using Wavelet Transform to Identify Singular Point of QRS Complex

3.6.1 Feature Extraction Using Wavelets

Most of the time, the important information of signal resides on the irregularities and singularities of the signal and wavelets can be used to extract that information. When the filter bank and wavelets are chosen appropriately, the wavelets are able to capture the irregularities and singularities of the signal. Mathematically, the local singularity of a signal is measured using Lipschitz exponent, the inflection points of signal $f(n)$ appear as extrema at $\frac{df(t)}{dt}$ and as zero crossing points at $\frac{d^2f(t)}{dt}$. Therefore, Mallat has suggested using a wavelet which is the first derivative of a scaling function.

3.6.2 Lipschitz Exponent

The functions which are infinitely differentiable are described as smooth or with no singularity [XCWX13]. If at some point, the function has noncontinuous derivative, then the point is known as the singular point. The Lipschitz exponent is a good application to measure the singularity of the point.

3.6.3 Relationship between Lipschitz Exponent and Modulus Maximum

Mallat has shown in his paper [MH92] that, all signals and noise in there may be completely represented by their singularities and singularities are generally referred in terms of Lipschitz exponents. If a signal is n times differentiable at time t_0, then its nth derivative is singular and it will be described as Lipschitz α where $\alpha > n$. If a signal is continuously differentiable at time t_0, then it is non-singular and has Lipschitz exponent 1.

Signals can have negative Lipschitz exponent as well. For example, many signals have singularities with positive Lipschitz exponents whereas, noise has a negative Lipschitz exponent. Therefore, having this mind, it makes it possible to separate a signal from noise if the singularities of noise can be detected and separated.

Generally, it is known that the singularity of a signal is directly proportional to the Lipschitz exponent. Therefore, as the transform scale increases, the wavelet transform modulus maxima will also get increases (Lipschitz exponent > 0) and similarly, it will decreases when Lipschitz exponent < 0. It can be seen that R wave in the original ECG signal appears as a pair of positive and negative extreme in the waveform which resulted after the decomposition of a wavelet transform.

3.7 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 considered only for the analysis.
- Attribute file (.atr): It contains the annotation information of the ECG signal, annotated by the doctors.

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.

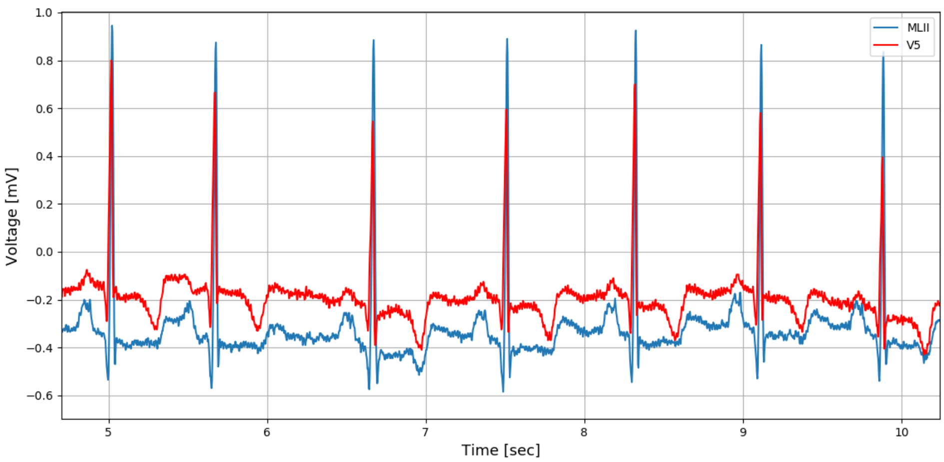


Figure. 3.3: The ECG signals from MIT-BIH dataset.

The signals are in a raw form and need to be processed before they can be used. Most of the time, the signals are also contaminated with noise, baseline drift, etc. and they are required to be cleaned to get the correct values.

3.8 Preprocessing

The ECG signal is required to be processed before it is analyzed, as it contains several noises and artifacts. The most common noises are the baseline wander and 60Hz power interference. Baseline wander generally appears because of the subject respiration or the body movements. It has a frequency range of 0Hz to 0.5Hz. The power interference affects the signals because of the electrical appliances in the surrounding.

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

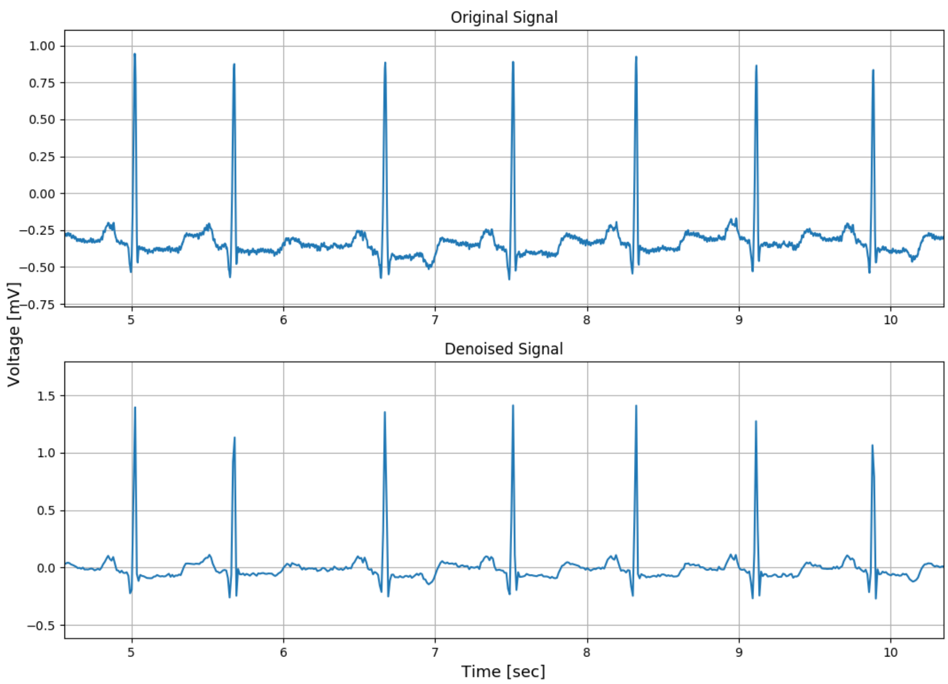


Figure. 3.4: The filtered ECG signal using wavelet.

3.8.1 Wavelet Transform Method

The wavelet transform is a very interesting technique for analyzing the signal in the time-frequency domain. It distributes the signal in such a way that the resulting block is well localized in both time and frequency. Decomposition of the signal into different frequency bands is obtained 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 output and the detail coefficients contain the high-pass filter output. The next step split the approximation coefficients again into 2 parts using the same process 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 can be seen in the Figure 3.4.

3.8.2 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. The range of frequencies, let say, f1 and f2, are called the frequency passband.

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 3 Hz to 45 Hz has been used. After passing the ECG signal through the band-pass filter, the resulting signal produced is the de-noised signal with no high-frequency and baseline drift. The de-noised signal can be seen in the Figure 3.5.

3.9 QRS Detection

QRS detection is the basis for processing the ECG signal. Regardless of what application is required, the accurate detection of QRS is a pre-requisite for feature extraction. A good wavelet base can help detect the features of ECG signal more appropriately with speed and accuracy. Therefore, Biorthogonal spline wavelet is used to detect QRS wave. Biorthogonal

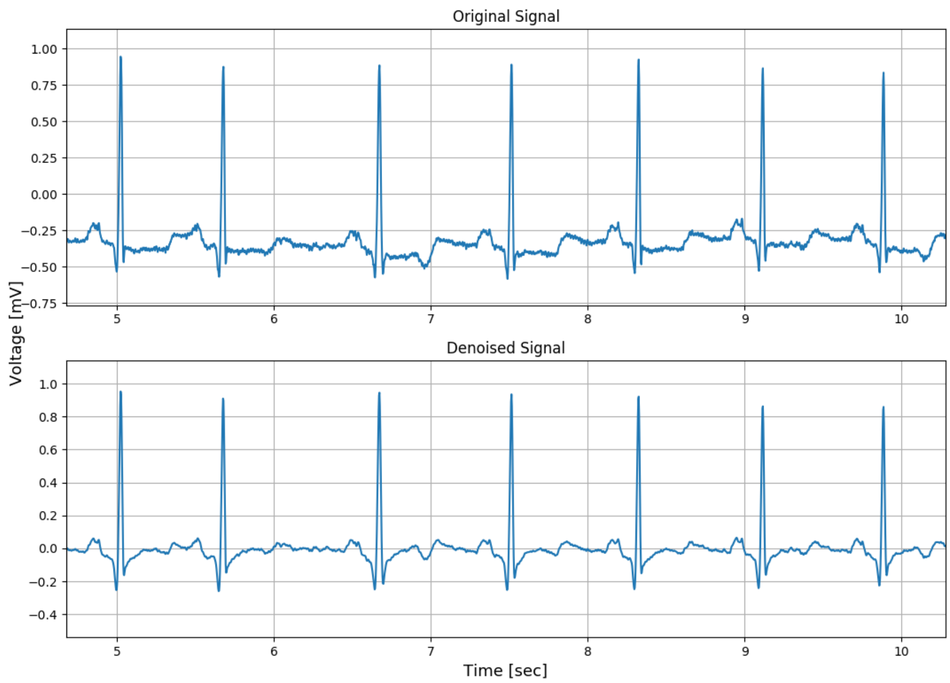


Figure. 3.5: The filtered ECG signal using bandpass filter.

spline wavelet transform of ECG signal is calculated using the Mallat algorithm. Figure 3.6 shows the Biorthogonal spline wavelet transform of ECG signal at 4 different scales.

Most of the QRS complex energy lies in the 3rd scale as can be seen in Table 3.4, therefore, the maximal minimal method can be used on the 3rd layer of the detail coefficient to find the R waves. The process starts by taking the first derivative of the 3rd layer to find the maximum and minimum points and then the 2nd derivative to locate the actual maximum and minimum values. The resulting waves are shown in Figure 3.7. As it can be seen that, there are other peaks available as well, therefore, to get the maximum and minimum pair only, a threshold needs to be set. And all the values that do not fulfill the threshold should be discarded. For finding the threshold, the result of the 2nd derivative is divided into 4 parts and from each part, the maximum value is located. After getting the values, the average is calculated for these values and that average is then divided by 3. The resultant value is a new threshold. The pair can be seen in Figure 3.8.

The value of R wave lies at the zero-crossing point (with a little delay) which is between the maximum and the minimum value pair. For compensating the delay, a maximum value can be searched in the window of 20 points to the zero-crossing point. The detected R-peaks can be seen in Figure 3.9. The flow chart for finding the R peaks is shown in Figure 3.10.

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

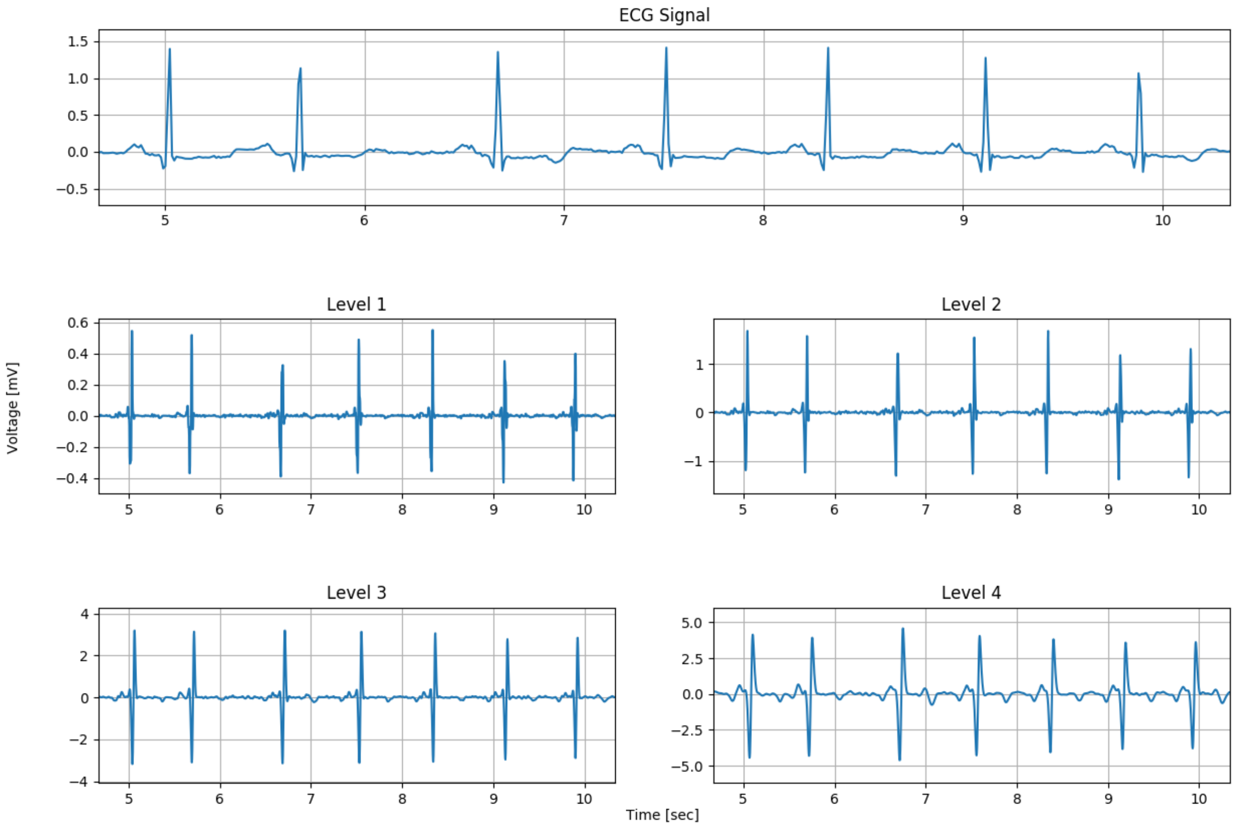


Figure. 3.6: ECG signal and its decomposition at different scales.

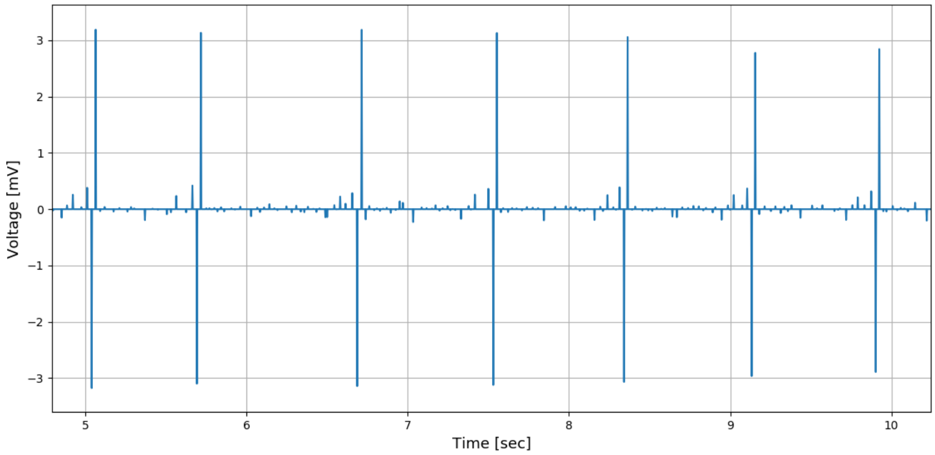


Figure. 3.7: Maximum and minimum values on scale 3.

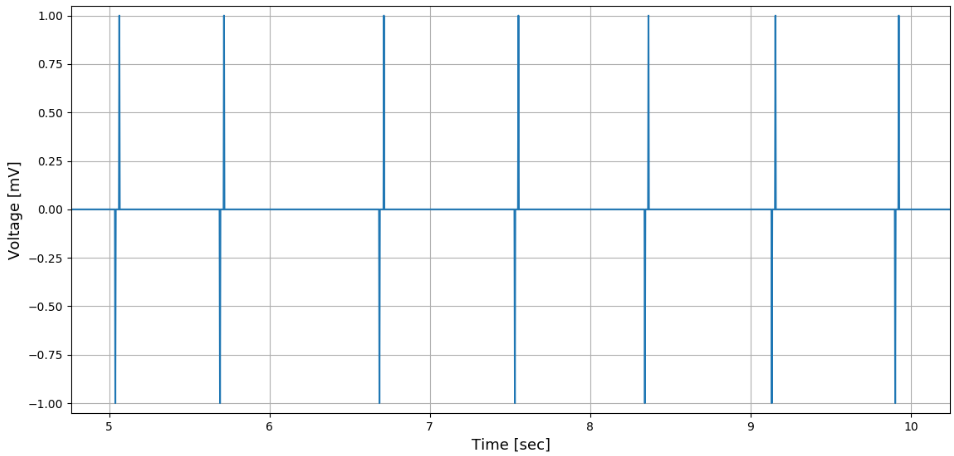


Figure. 3.8: Maximum and minimum values pair for finding the R peaks.

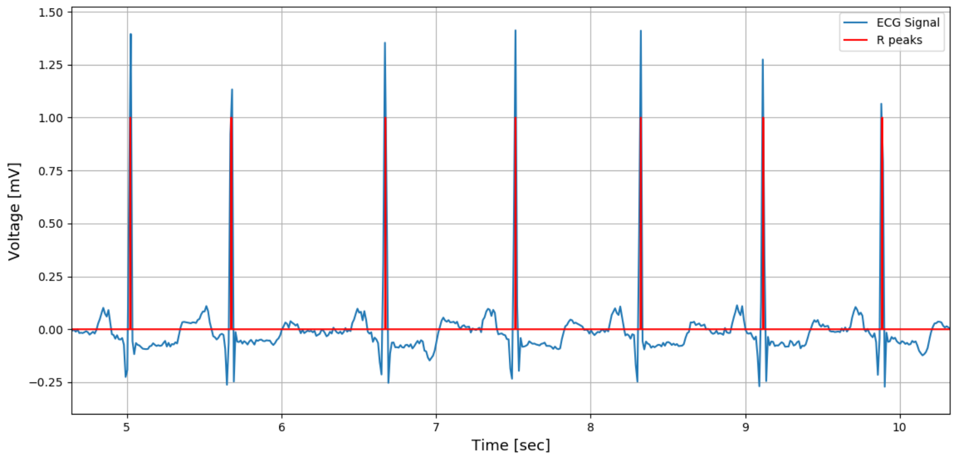


Figure. 3.9: Detected R peaks.

After getting the R-peaks, Q and S peaks have to be detected. Q and S wave generally are of high frequency; therefore, their energies are mainly on 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 3.11.

After detecting the QRS complex, P and T waves are required to be detected as they also have very important significance to identify the arrhythmia. P wave generally occurred before the QRS complex and T wave after the QRS complex, therefore, they can be detected based on QRS location.

3.10 P and T Wave Detection

Most of the P and T waves energy lies on the scale 4 and the QRS complex energy lies on the scale 3 of detail coefficient. If QRS complex (that was detected on scale 3) is used, it sometimes misses the P wave or identifies the wrong position. Therefore, it is first required to detect QRS complex on scale 4 and then find P and T waves. The same algorithm is used to detect the QRS complex on scale 4 that is used to detect on scale 3, as described in section 3.9.

After getting the QRS on scale 4, a window size of 100 is used for detecting the P wave. The starting point of the window is one sample less than from Q wave position and if the window size is added to this position, we get the beginning of the window. RR interval

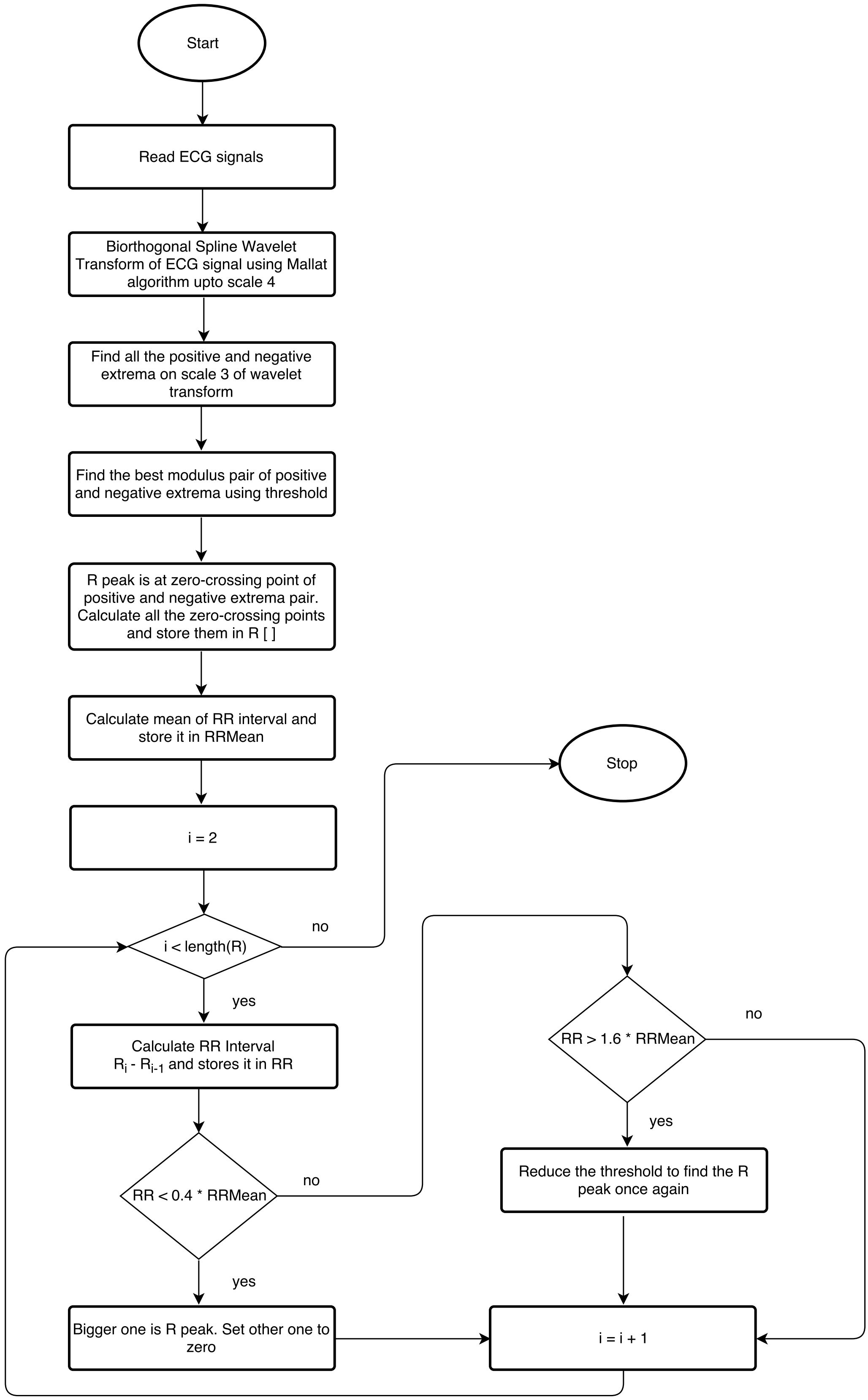


Figure. 3.10: R peak flow chart.

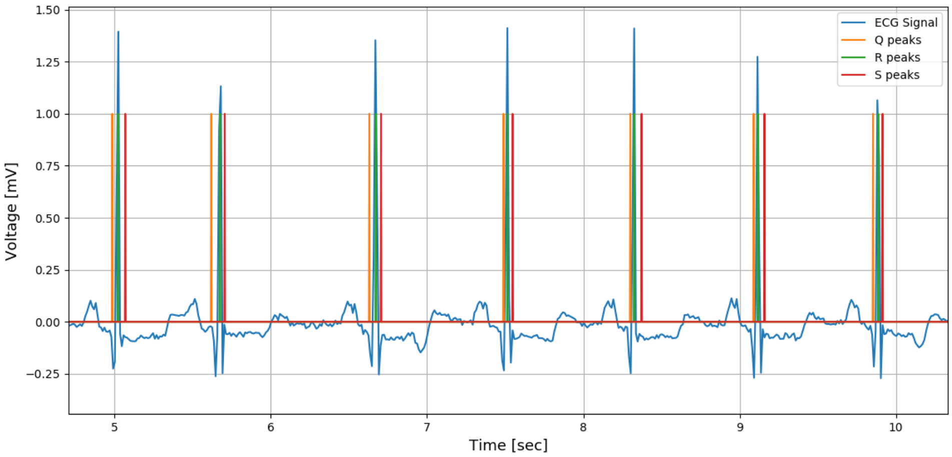


Figure. 3.11: Detected Q, R and S peaks.

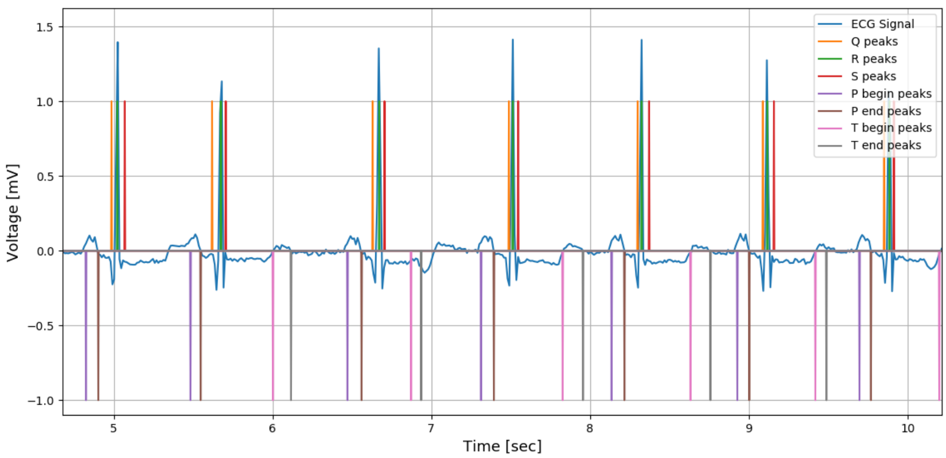


Figure. 3.12: Detected P,Q,R,S and T waves.

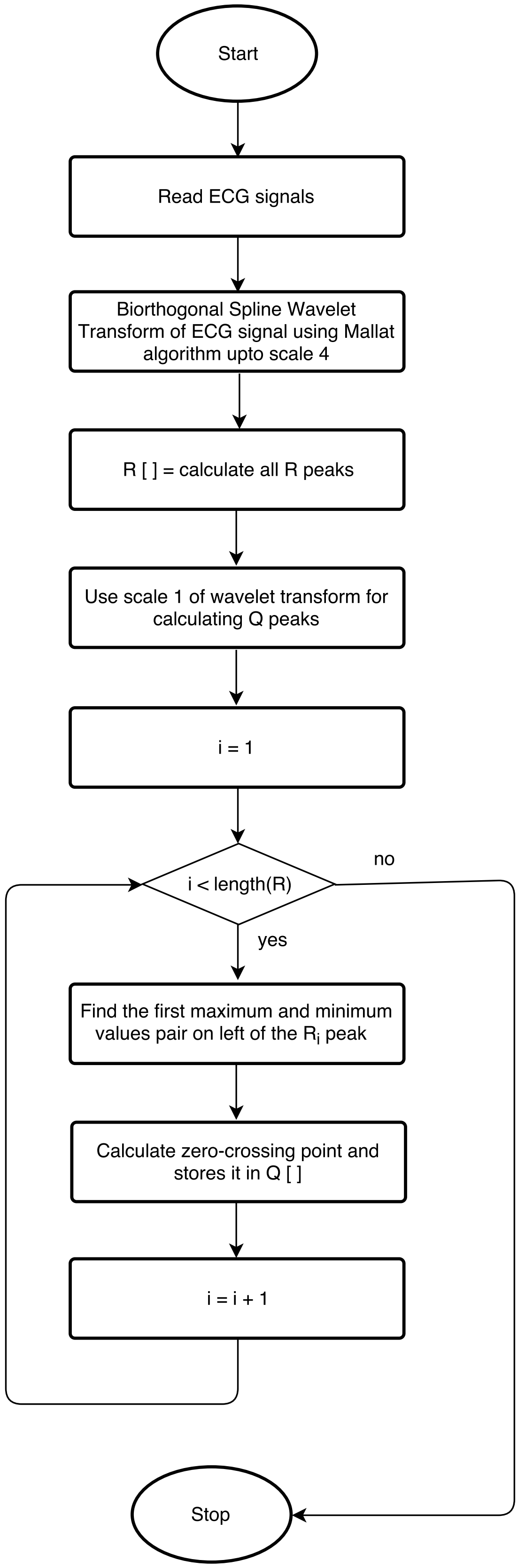


Figure. 3.13: Q and S peak flow chart.

is also calculated between the 2 consecutive R peaks. 1/3 of the RR interval is used for detecting the P wave and 2/3 of the RR interval is used for detecting the T wave. Once the window is identified on the scale 4, the max and min values are identified in that window as P wave lies on the zero-crossing point of min and max pair. The average is taken to calculate the P wave position of scale 4. Once the P wave position is calculated, the P wave is identified relatively on the original signal. One point to note here is that the scale 4 data is shifted because of filtering, therefore, it is required to shift the detected position few samples back to get the appropriate value.

The same approach is used for detecting the T wave, but instead of looking the window before the QRS complex, here the window is searched after the QRS complex for detecting the T wave.

All detected waves can be seen in Figure 3.12. The flow chart for finding the P wave and T wave is shown in Figures 3.14 and 3.15 respectively.

3.11 Heart Rate Calculation

The heart rate is calculated by first counting the number of R peaks in a 10 seconds window and then multiply the count by 6 to get the heart rate.

$$HeartRate = number\_of\_R\_peaks \times 6$$

3.12 Algorithm Execution on ECG Chair Data

The data used in the development of ECG feature extraction algorithm is taken from MIT-BIH data set and the algorithm works great with that data set. To analyze the performance of algorithm on the ECG chair data that is used in the implementation of the system, the data is collected from the ECG chair and passed to the algorithm. It turns out that the algorithm works perfectly fine with this data as well. The original ECG signal can be seen in Figure 3.16 and the extracted features can be seen in Figure 3.17.

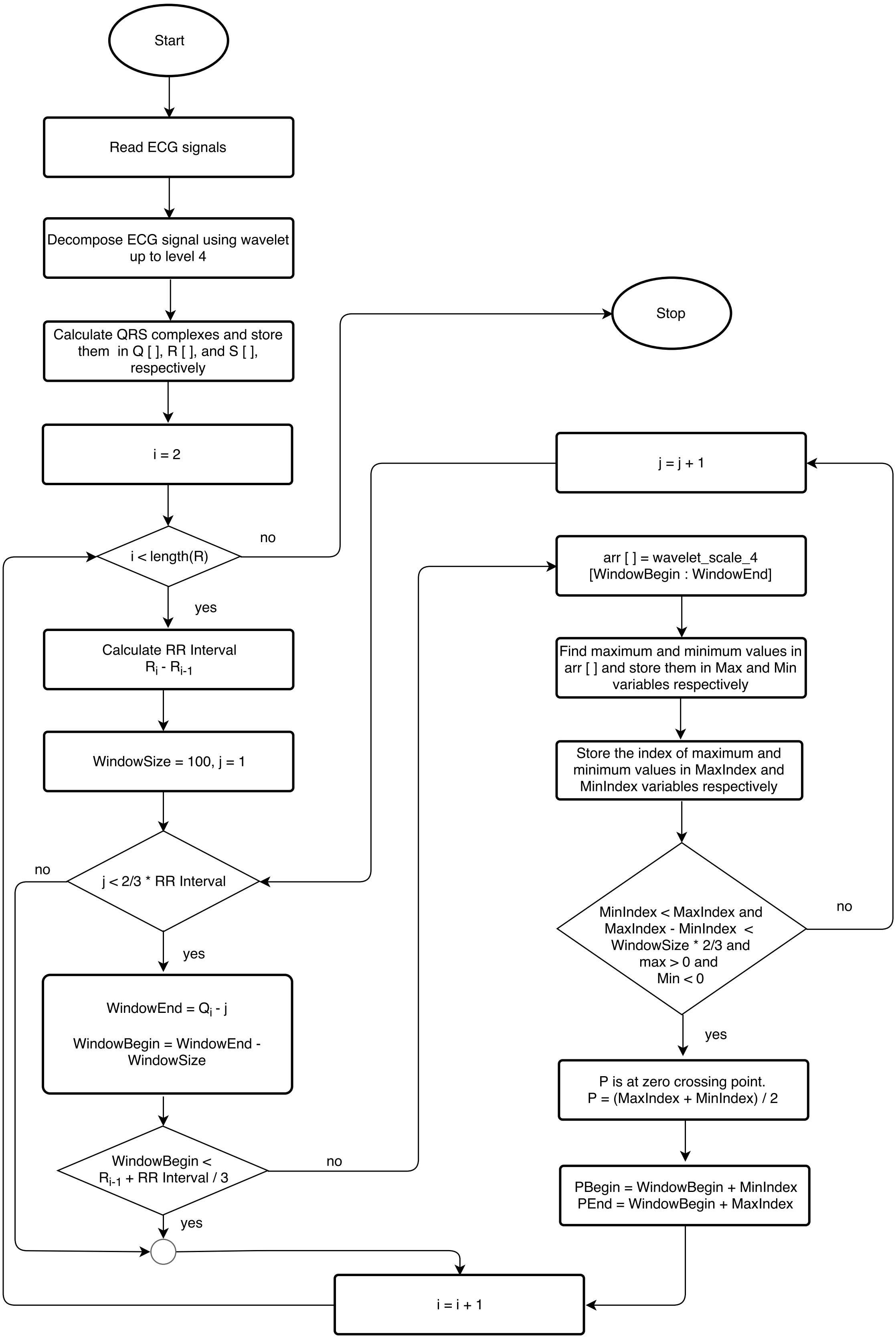


Figure. 3.14: P wave flow chart.

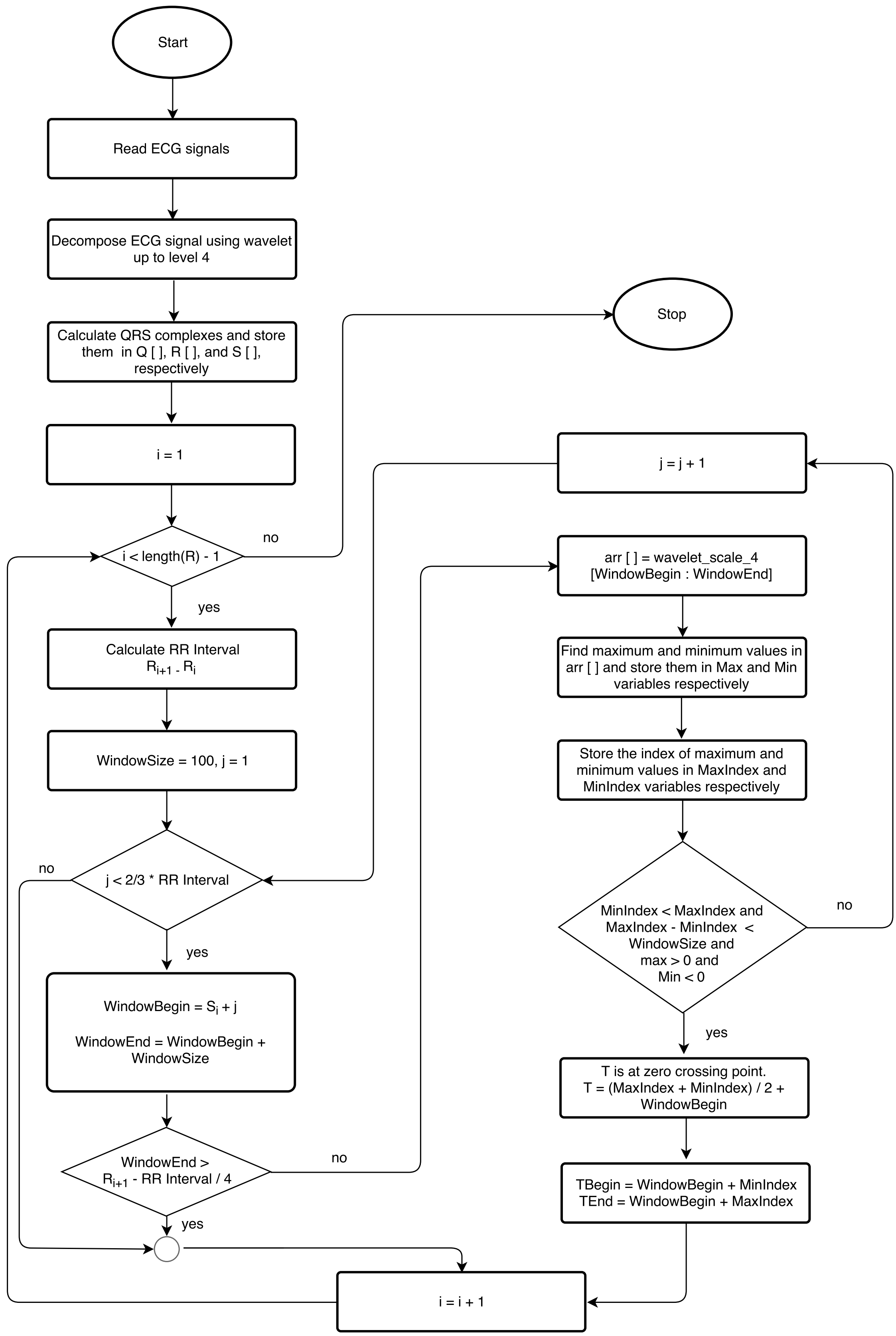


Figure. 3.15: T wave flow chart.

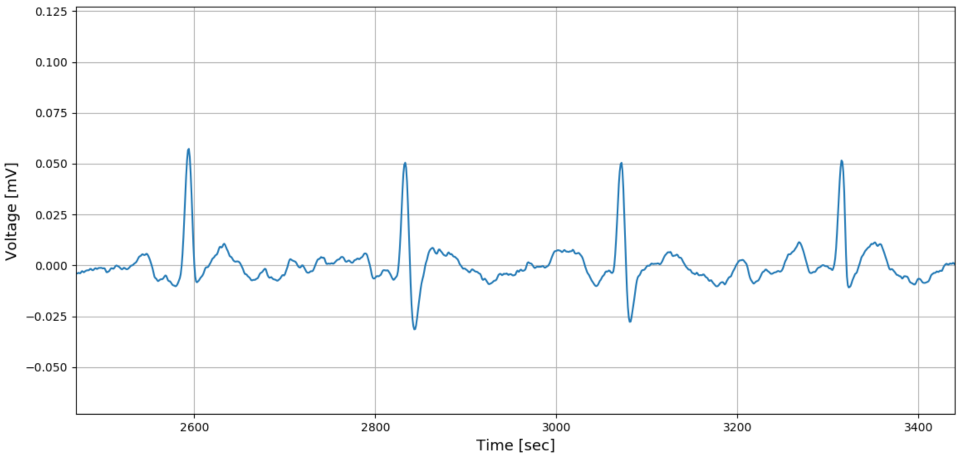


Figure. 3.16: Original ECG chair signal.

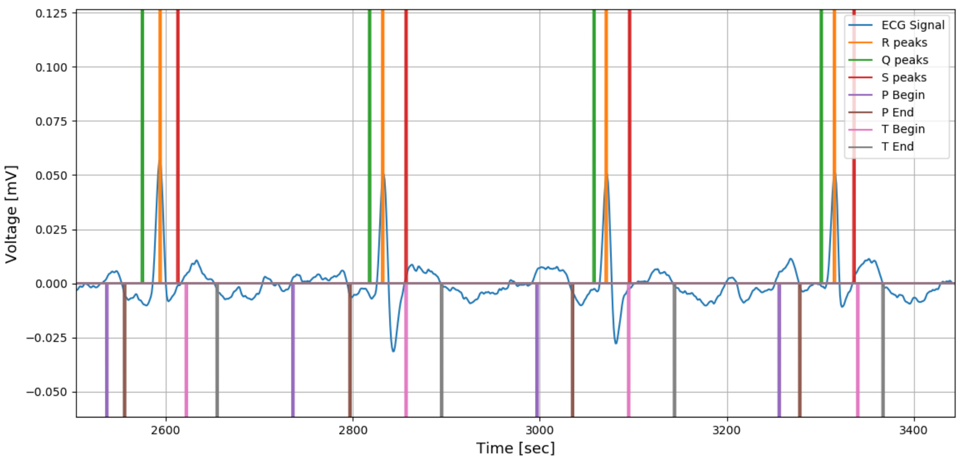


Figure. 3.17: Detected P,Q,R,S and T waves on the ECG chair signal.

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. It has been used in many technologies such as, in medical to identify the diseases, automatic game playing, self-driving cars, image recognition, natural language processing and many more. 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 field "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].

1. Supervised Learning

2. Unsupervised Learning

4.1.1 Supervised Learning

In supervised learning, the labeled data is used to train the models. Here, labeled data represents that the input and output variables 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 value is 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 do the predictions. The workflow can be seen in Figure.

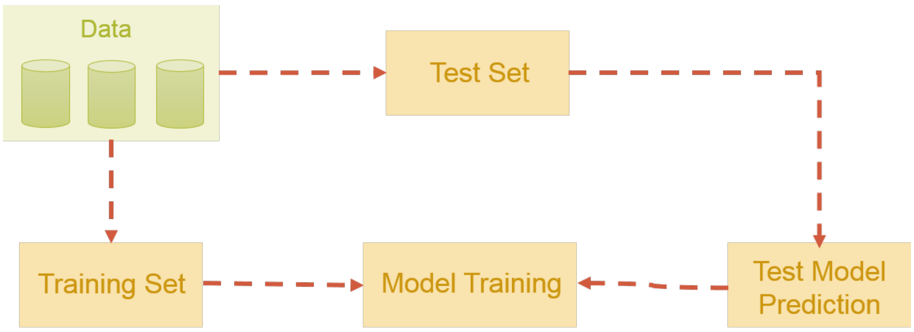


Figure. 4.1: Basic supervised machine learning workflow.

Machine learning is generally powered by a huge amount of data which is generally referred aa 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 making a correct 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, 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

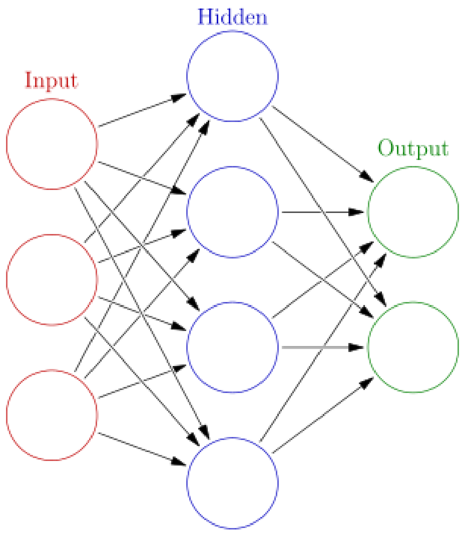


Figure. 4.2: An artificial neural network with its 3 basic layers[con17a].

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 weights, the more important they are. 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. An activation function is a function which takes an input an generates an output based on a certain threshold. So, if the summed value is greater than the threshold value of that activation function, the output is produced or in other terms, the neuron fired else no output is produced and neuron does not fire.

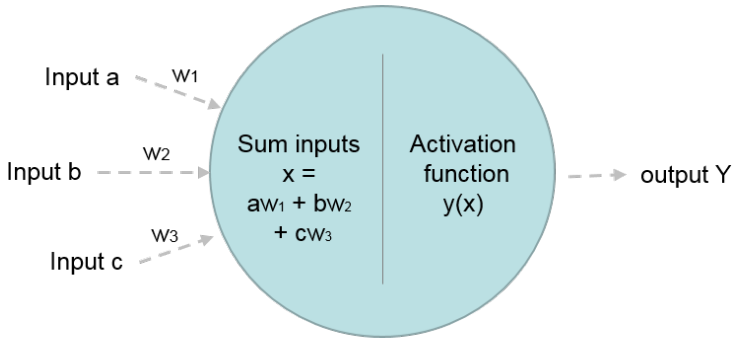


Figure. 4.3: A single artificial neuron.

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:

$$y = \frac{1}{1+e^{-x}}$$

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 as it is the ideal choice to be used in convolutional neural networks. It takes the real value input and all negative values are mapped to zero. It is represented as:

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

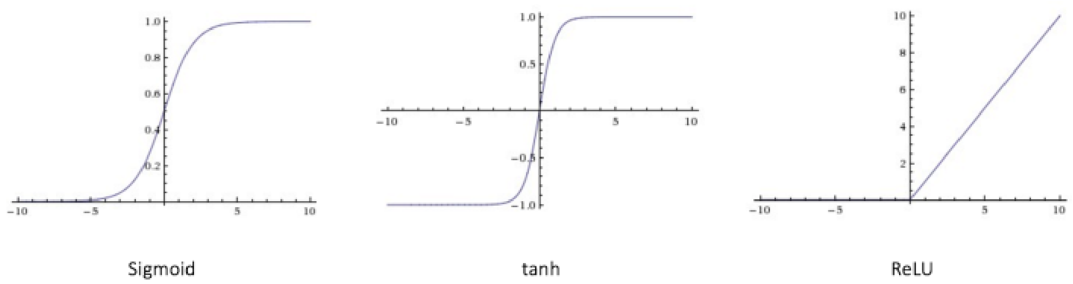


Figure. 4.4: Activation functions[ujj17].

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

4.2.3 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.

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

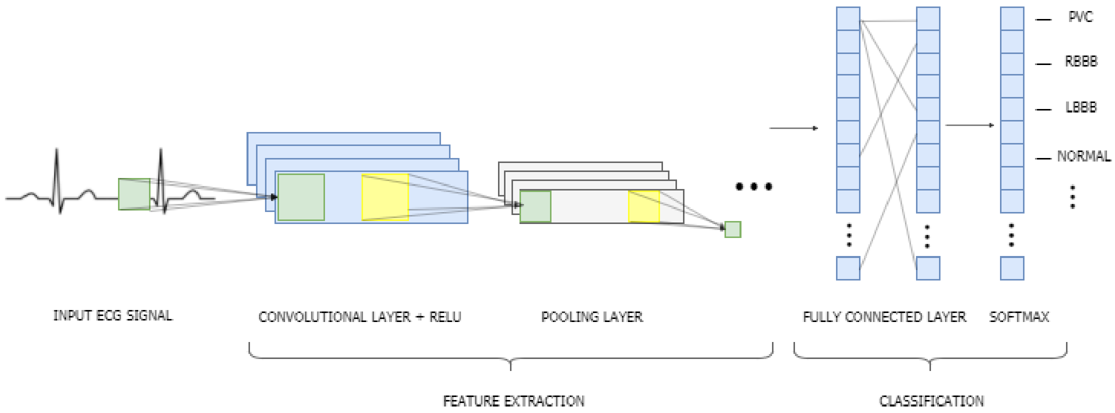


Figure. 4.5: Convolutional neural network example.

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 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.

Pooling Layer

Pooling layer is used to downsampled the convolutional layer output. There are several pooling options, for example, average pooling, L2-norm pooling, etc., but max polling 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 can be seen in Figure 4.6.

Fully Connected Layer

In fully connected layer, the neurons of one layer are connected to all the neurons of the other second layer, as it can be seen 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

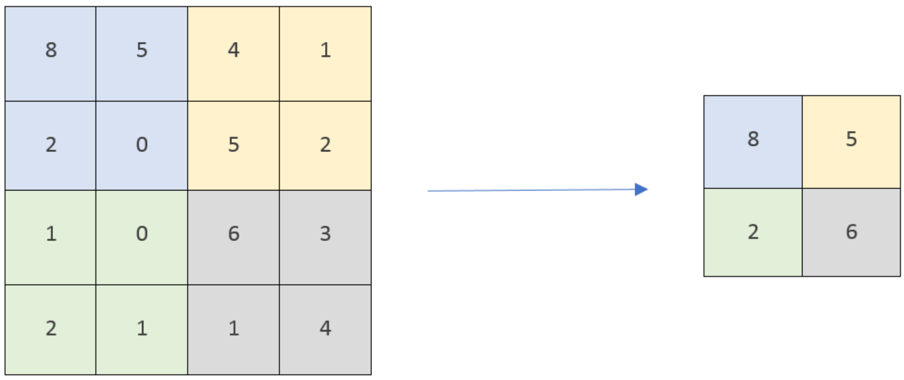


Figure. 4.6: Max pooling layer example.

to produce the probabilities for the classification of the input.

4.2.4 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 was 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: To optimize the neural network, for example, rse, adagrad, adam, etc.
- Loss Function: This is the value that model tries to minimize to calculate the error, for example, categorical_crossentropy, mse, etc.
- Metrics: It can be any existing metric or a custom defined metric function. But for classification problems metrics=['accuracy'] is recommended.

Then the model is trained by using the *fit()* method. This method lets the model iterate over the data and find the most optimal neural network for the given data.

This library has been used for training the CNN for the identification of Cardiac Arrhythmias.

4.2.5 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 can be seen in Figure 4.7.

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. For both convolutional layers, Relu has been used as an activation function each followed by a MaxPooling layer. The batch size of 1000 was used for the training, along with the Adam algorithm to optimize the CNN. Since the model is trained for performing the classification of an ECG signal, therefore, categorical cross-entropy loss function was 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 classes.

The model was trained and tested for the several no. of iterations ranging from 10 to 100. The best model was found after the 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.

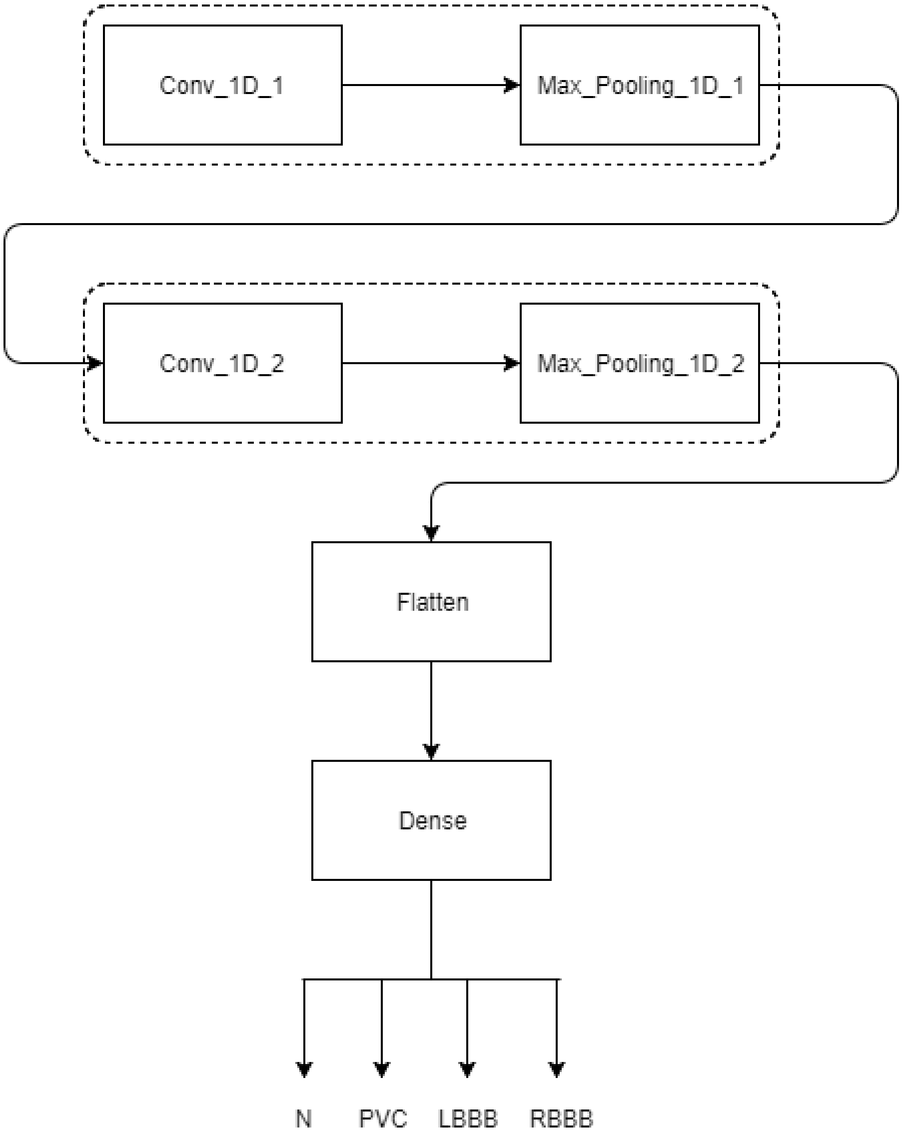


Figure. 4.7: A 6-layer Convolutional Neural Network model for the identification of cardiac arrhythmia.

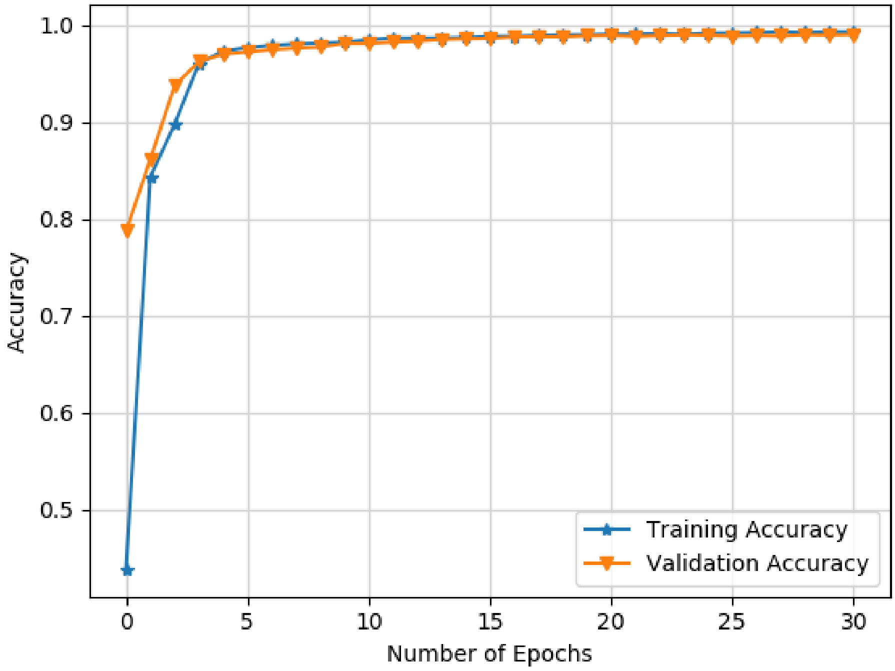
```
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
conv1d_1 (Conv1D)            (None, 300, 64)           256
_________________________________________________________________
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
=================================================================
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.

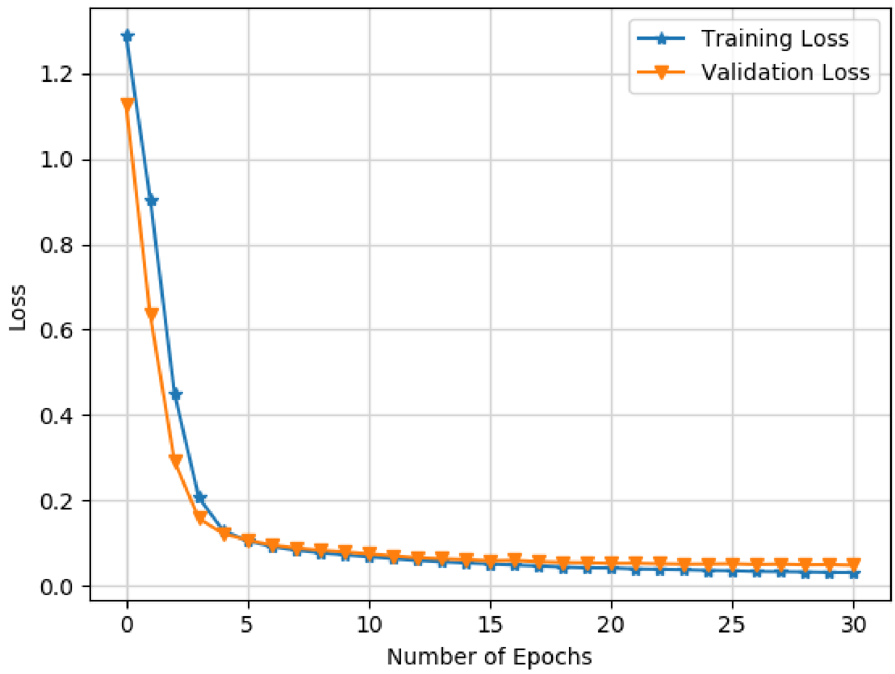
Results

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 table, $a \rightarrow b$ means is that a was classified as b.



(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**

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

"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 through out 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, weather reports etc. It can connect to many data sources and pull data from it to do the analytics or 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 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 s, ms, μs, ns. If the field does not contains 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. It is so simple that, the setup can be done in just 5 minutes. First, a user needs to create a database in the 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, go to *localhost:3000* address to open the Grafana web interface. Select an option to add a new data source. Lets say the database name is “medit”, then configure the InfluxDB data source as shown in Figure 5.1.

Once the data source is setup, 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, alert list, etc.

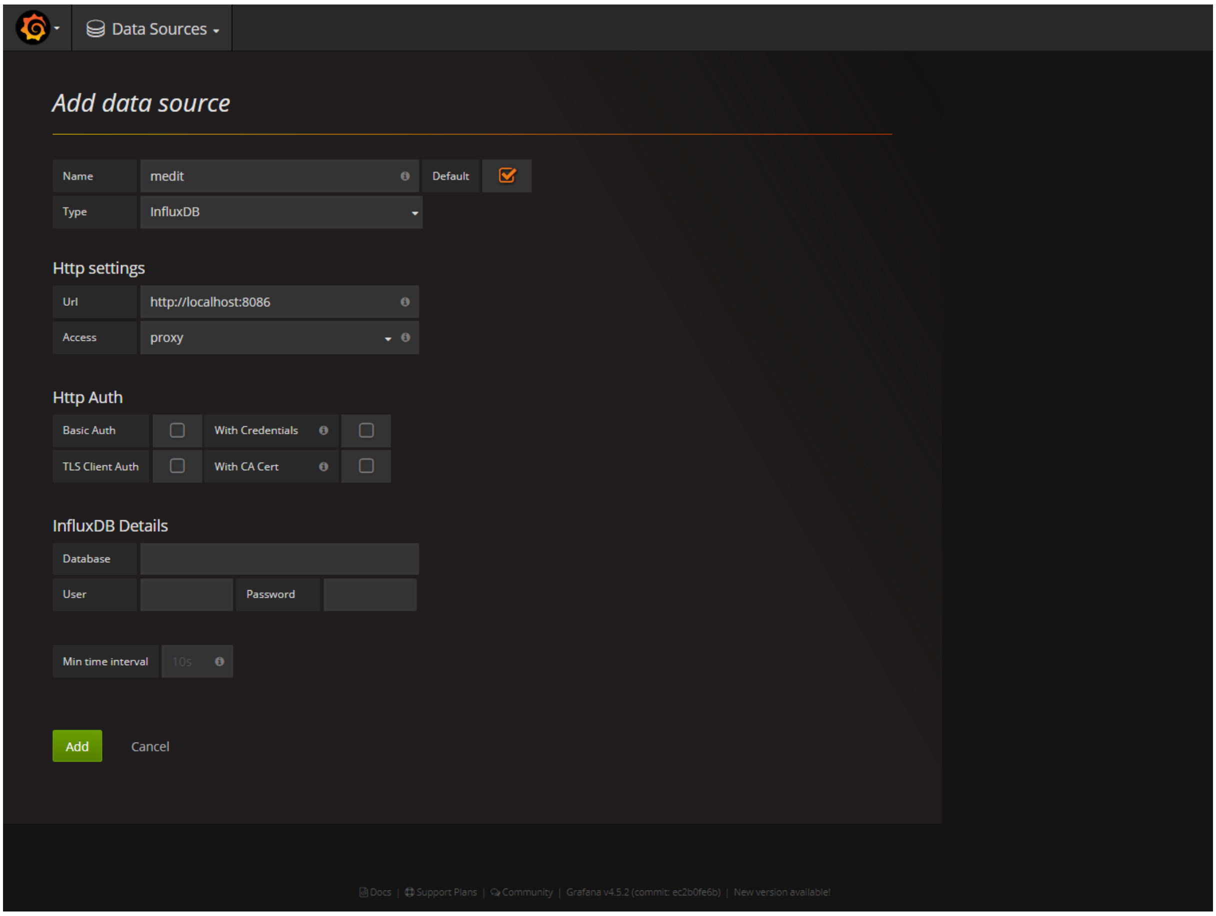


Figure. 5.1: InfluxDB data source setup in Grafana.

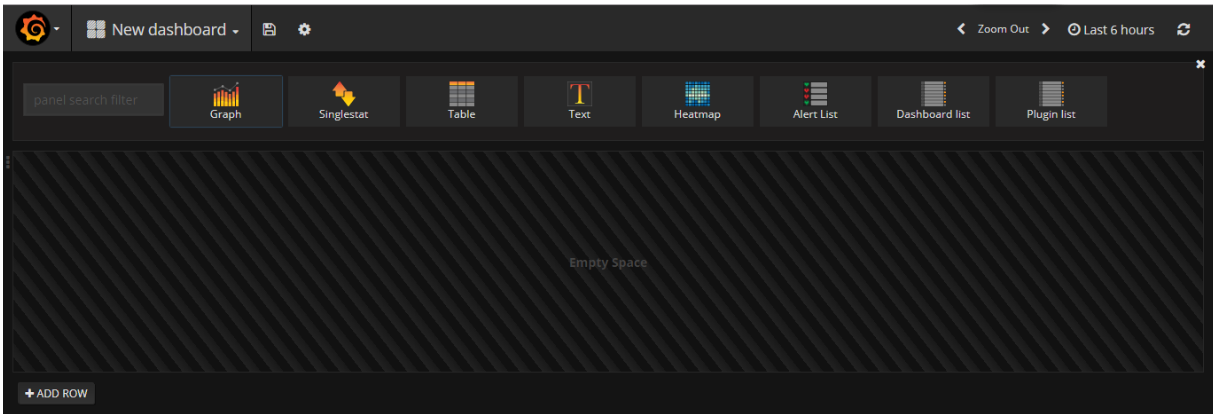


Figure. 5.2: Dashboard setup.

Lets say a graph is added to the dashboard. Now, a data source is need 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 graph can be seen in Figure 5.3. A very user-friendly interface is available where one can define a query for the panel. A query can be build just by selecting the option from drop down. It will contains all the information from the database that is setted up in the data source, that shown in Figure 5.1. So 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 graph. Once all the steps are done, the panel is ready to display the information from the InfluxDB.

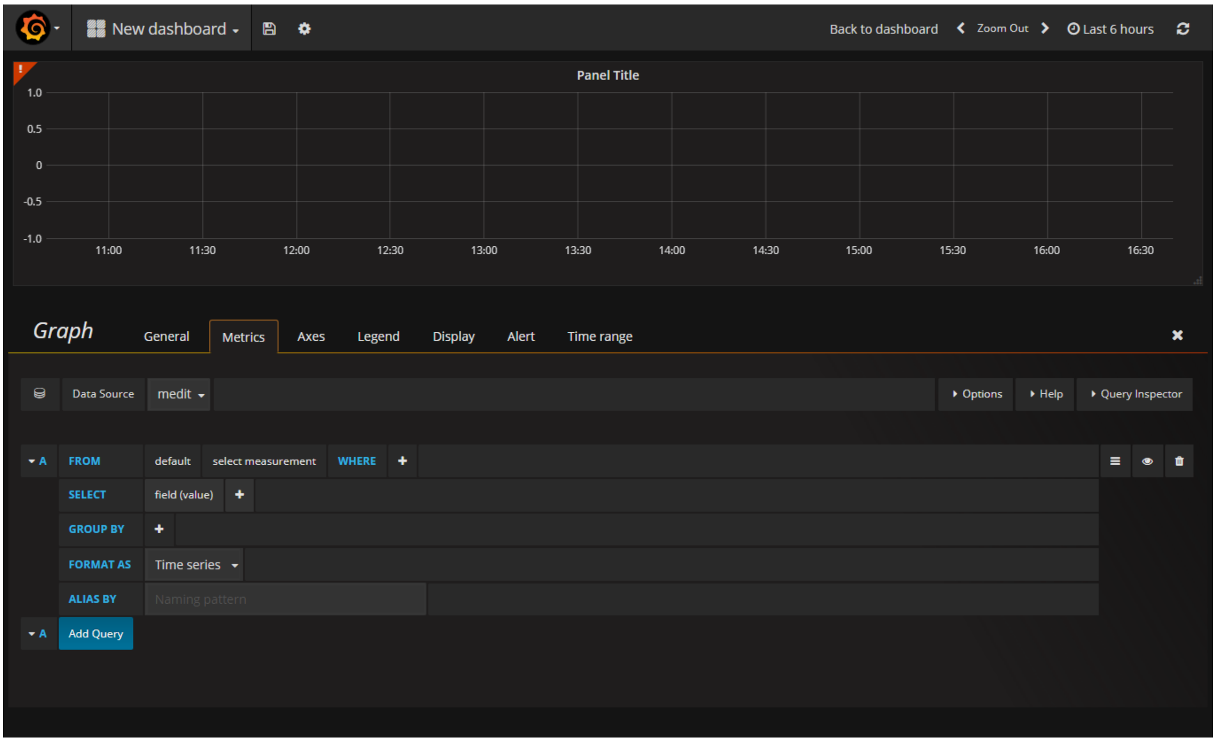


Figure. 5.3: Interface to add query for the panel.

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 web logs, events, social networks, sensors data,etc, 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 ensure 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:

1. Speed Layer
2. Batch Layer
3. Serving Layer

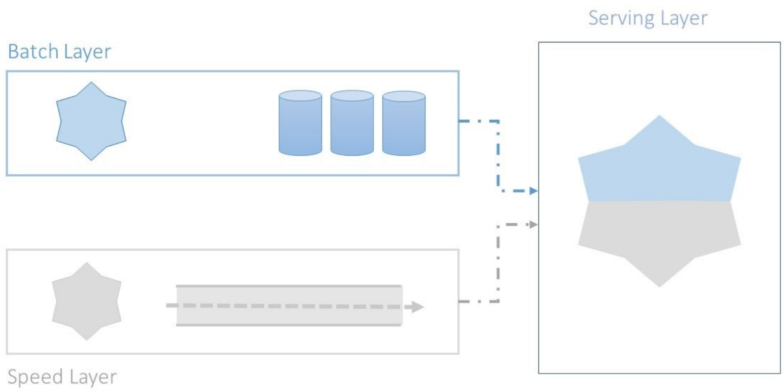


Figure. 5.4: Lambda architecture.

5.4.1 Batch Layer

The Batch layer will hold all of the master data which will be stored in Apache Hadoop. This data will be kept in its original state, untouched and in an immutable manner. This data will be processed and generate the batch views which then will be served in the serving layer. This is the place which will provide 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 the speed layer.

Speed layer process the real-time stream and generate the real-time views of a short time frame, which are then serve 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 which can be used in this layers are Cassandra, HBase, etc.

The data source will provide the data which will be streamed in the stream layer and at the same time to the batch layer. Batch layer will hold the data for a long time and stream layer will process the stream in a window of short time and then provide the calculated result. The serving layer will combine 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 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.

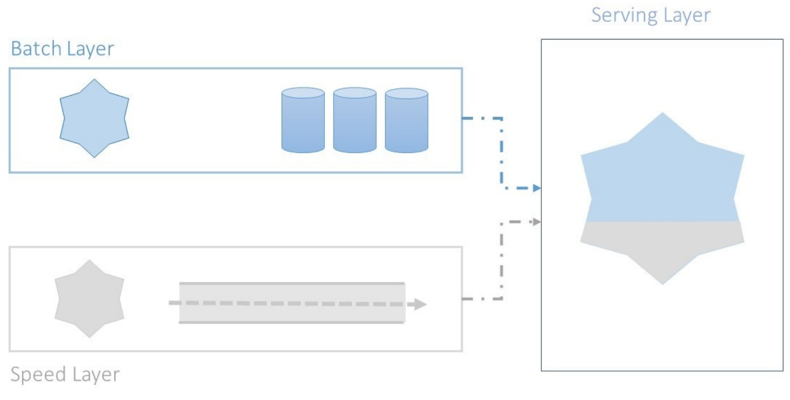


Figure. 5.5: Using more data from serving layer in case of speed layer having a problem.

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

Figure. 5.6: Lambda symbol representing the lambda architecture.

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 to hold. In contrast, the data will be removed from the queue, once it is delivered to the appropriate consumer of the data. It uses topics to publish and to subscribe for the data. Every Kafka server is known as broker and since it is a distributed system, therefore, there can be more than one broker. The more the number of brokers are, the higher the availability is. The advantage of having Kafka is that it can handle many different forms of data, such as sensors data, application events, server logs, social network events etc. Kafka is very fast despite of having heavy load.

The implementation of lambda architecture has been shown in Figure 5.7. The data will directly be published from 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 where the data publish to. 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 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 interface for several languages to write the applications. The available languages are Java, Scala, Python and R.

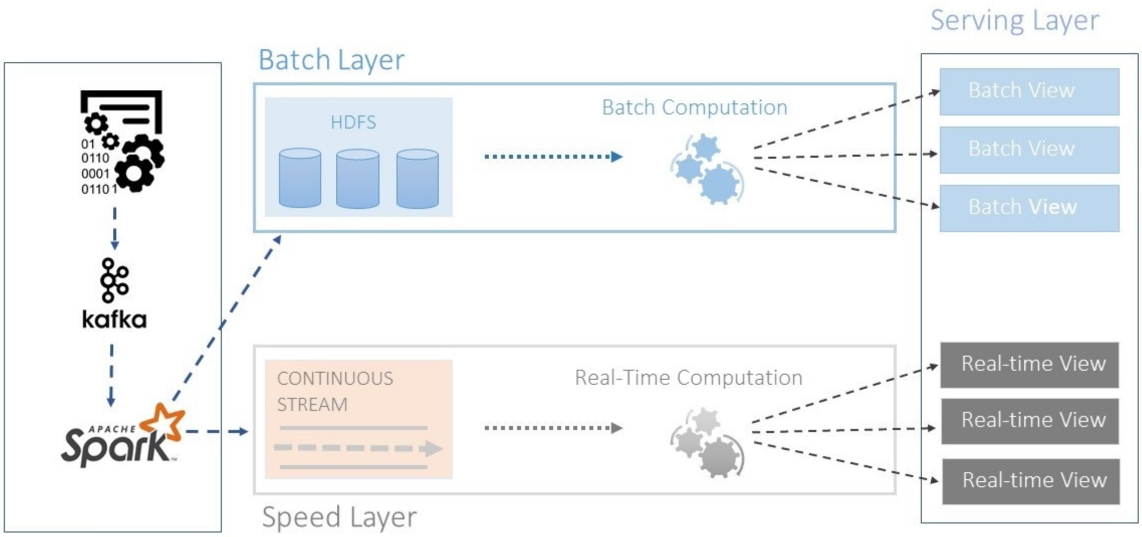


Figure. 5.7: Implementation of lambda architecture.

5.5 System Architecture

A system has been implemented to collect, process and visualize the data for all the sensors based on the lambda architecture defined in Section 5.4. But the system has been modified according to the requirements and because of the lack of enough resources to run all the big data tools. The system architecture can be seen in Figure 5.8 .

An architecture has been implemented in *pytho* language, which first collects the data from all sensors based on their protocol. The data is then immediately stored in InfluxDB in order to have the original state of the data, as shown in Figure 5.9. The data is then cleaned and processed to do the further operations on it, such as for the ECG signal, the feature extraction.

The main focus of this thesis is on cardiac, therefore, most of the implementation is based on ECG sensor. The NI USB-6259 is used for the signal acquisition. The sample rate of 360 Hz has been used as the deep learning model was trained on 360 Hz frequency data. The same feature extraction algorithm is used that is defined in Section 3.9. Once the R wave is detected, 100 points to the left and 150 points to the right of the R peaks are selected to make a single ECG signal of 250 sample points, which was then passed to the deep learning model to predict the arrhythmia. The window of 10 seconds is used for calculating the heart rate. The complete process is shown in Figure 5.9. Once the features are extracted, the cleaned signal and the extracted features are saved in InfluxDB immediately to save the state of the signal.

Once all the information is stored in the InfluxDB, then the Grafana takes the job. It retrieves the desired data from InfluxDB and display it on the dashboard. Grafana provides many other options as well, such as, the refresh time, i.e, how much time retrieves the new data from InfluxDB. If the data is coming from other machine and the clock time is not same, Grafana gives an option to shift the time to adjust the panels and many other helpful functions.

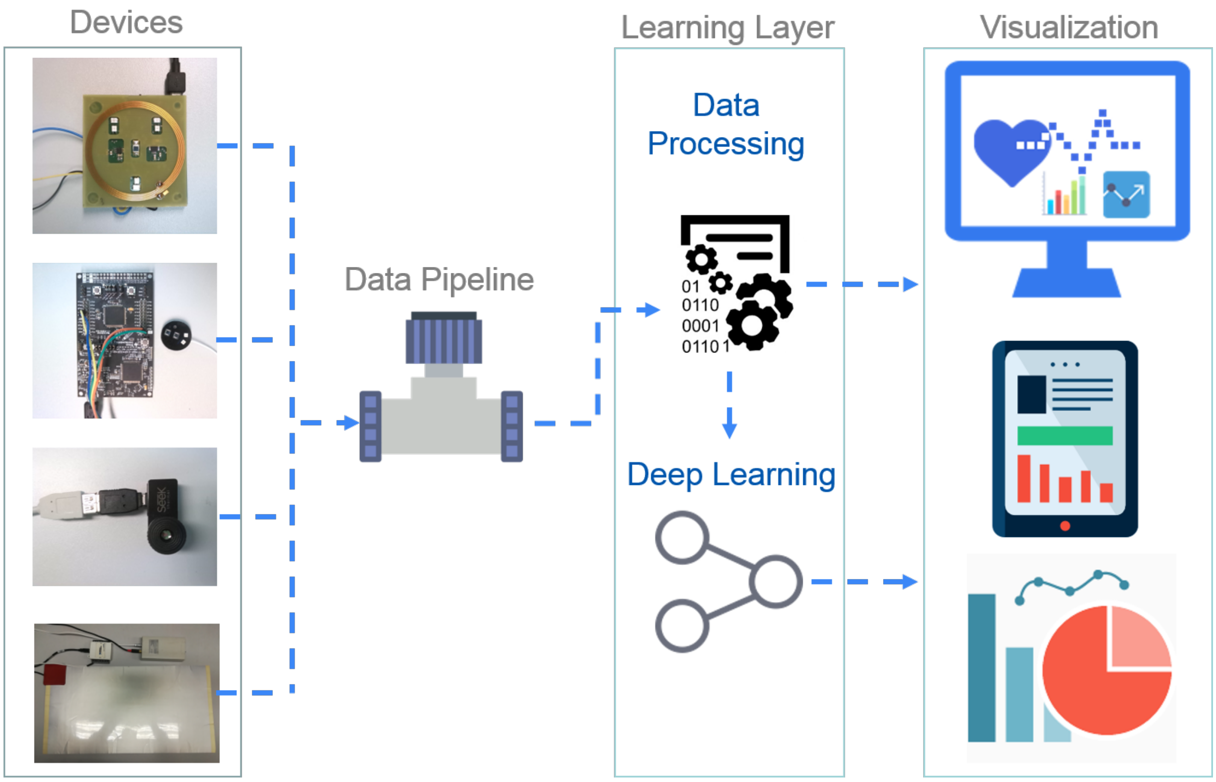


Figure. 5.8: System architecture for the sensors.

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

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 start, stop and in case of error, the logs can be viewed directly in the same interface. The interface pictures can be seen in Figure 5.13.

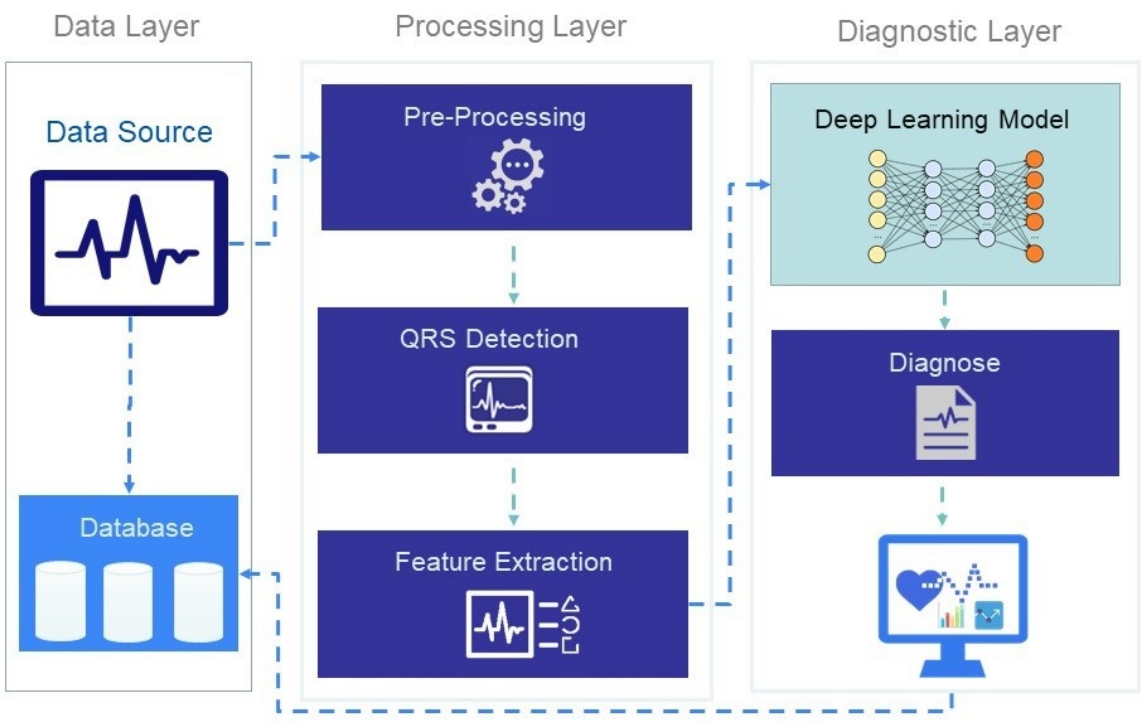


Figure. 5.9: Using the deep learning model in real-time environment.

Figure. 5.10: Real-time ECG signal visualization.

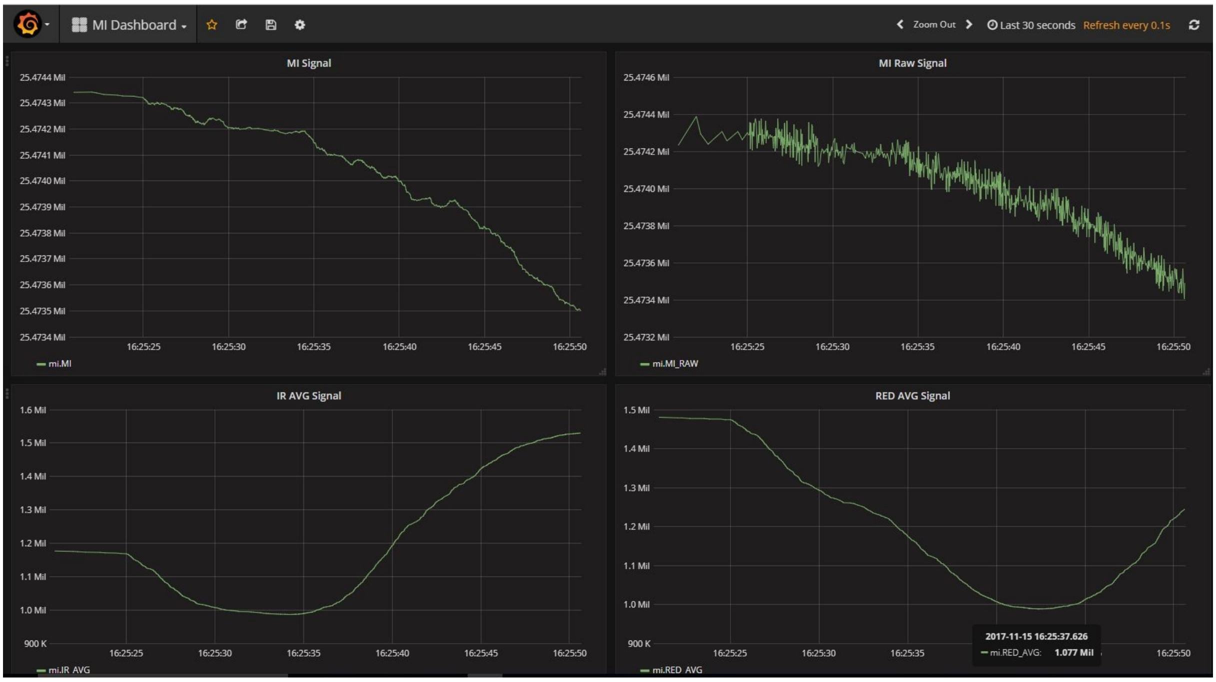


Figure. 5.11: Real-time MI signal visualization.

Figure. 5.12: Real-time PPG signal visualization.

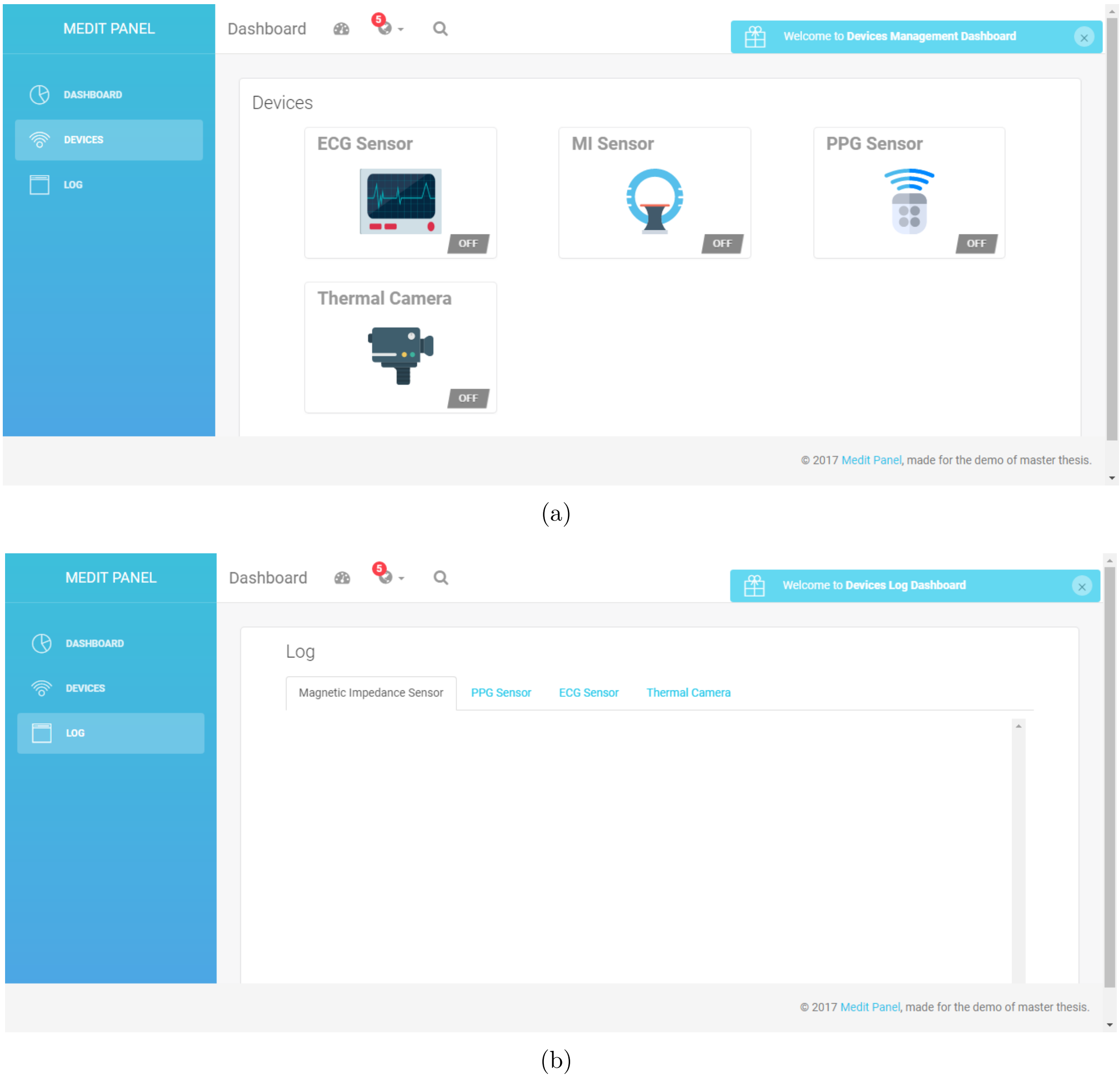


Figure. 5.13: (a) The device management view to interact with devices; and, (b) the log view to know the state of each device.

5.7 Android Application

The purpose of this thesis is to implement a system which allows to visualize the data from anywhere. Sometimes, it is not very convenient to visualize the data on laptop, therefore, an android application has been written so the sensors data can be visualized directly on your mobile device as well. The advantage of android application is that it is very light weight and does not require any dependency to visualize the data. The screen shot of android application can be seen in Figure 5.14.

Figure. 5.14: Real-time ECG signal visualization on android application.

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

[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=kTNoQgAACAAJ. – 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]

[Com17a] Commons, Wikimedia: *File:Brantigan 1963 1-53.png — Wikimedia Commons, 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

[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]

[Iye17] Iyer, Sulakshana: *Why is data visualization important.* https://www.quora.com/Why-is-data-visualization-important. Version: 2017. – [Online; accessed 29-December-2017]

[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.

[mau17] mauvila: *ECG artifcats.* http://www.mauvila.com/ECG/ecg_artifact.htm. Version: 2017. – [Online; accessed 12-November-2017]

[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=mWtHAAAAQBAJ. – 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

[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

[MS14] MEENA, Ravi S. ; SHARMA, Ambalika: COMPARISON AND ANALYSIS OF ORTHOGONAL AND BIORTHOGONAL WAVELETS FOR ECG COMPRSSION. In: *Int. J. Res. Eng. Technol.* (2014), Nr. 3

[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. – DOI https://doi.org/10.1016/0031–3203(91)90083–H. – ISSN 0031–3203

[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. – ISSN 0018–9294

[PZZ10a] 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

[PZZ10b] PAN, Tao ; ZHANG, Lei ; ZHOU, Shumin: Detection of ECG characteristic points using biorthogonal spline wavelet. In: *Biomedical Engineering and Informatics (BMEI), 2010 3rd International Conference on* Bd. 2 IEEE, 2010, 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. ; Nissila, 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. – 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]

[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

[sum17] sumdu: *arrhythmia*. https://elearning.sumdu.edu.ua/free_content/lectured:a3664646bdd2fb5e12ac4d9fc4d3dff83793be56/20150113065426//64025/index.html. Version: 2017. – [Online; accessed 13-December-2017]

[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

[XCWX13] Xing, S. ; Chen, S. ; Wei, Z. ; Xia, J.: *Unifying Electrical Engineering and Electronics Engineering: Proceedings of the 2012 International Conference on Electrical and Electronics Engineering*. Springer New York, 2013 (Lecture Notes in Electrical Engineering). https://books.google.de/books?id=plq4BAAAQBAJ. – ISBN 9781461449812

[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

[ZLSJ08] Zheng, Xiuyu ; Li, Zhen ; Shen, LinLin ; Ji, Zhen: Detection of QRS complexes based on biorthogonal spline wavelet. In: *Information Science and Engineering, 2008. ISISE'08. International Symposium on* Bd. 2 IEEE, 2008, S. 502–506