Myocardial Infarction Detection Using ECG Signal Applying Deep Learning Techniques - ConvNet, VGG16, InceptionV3 and MobileNet

by

Samanta Tabassum Promita
17201132
Simon Abhijet Biswas
17201066
Nisat Islam Mozumder
17301067
Mamur Taharat
17101282

A thesis submitted to the Department of Computer Science and Engineering
in partial fulfillment of the requirements for the degree of
B.Sc. in Computer Science and Engineering

Department of Computer Science and Engineering
BRAC University
January 2022

Declaration

It is hereby declared that

1. The thesis submitted is our own original work while completing degree at BRAC University.
2. The thesis does not contain material previously published or written by a third party, except where this is appropriately cited through full and accurate referencing.
3. The thesis does not contain material which has been accepted, or submitted, for any other degree or diploma at a university or other institution.
4. We have acknowledged all main sources of help.

Student's Full Name & Signature:

Nisat

Nisat Islam Mozumder
17301067

SAMANTA TABASSUM

Samanta Tabassum Promita
17201132

Taharat

Mamur Taharat
17101282

Simon

Simon Abhijet Biswas
17201066

Approval

The thesis titled "Myocardial Infarction Detection Using ECG Signal Applying Deep Learning Techniques" submitted by

1. Samanta Tabassum Promita (17201132)
2. Simon Abhijet Biswas (17201066)
3. Nisat Islam Mozumder (17301067)
4. Mamur Taharat (17101282)

Of Fall, 2021 has been accepted as satisfactory in partial fulfillment of the requirement for the degree of B.Sc. in Computer Science and Engineering on January 20, 2022.

Examining Committee:

Supervisor:
(Member)

Dr.Jia Uddin
Assistant Professor
AI and Big data Department
Endicott College
Woosong University

Co-Supervisor:
(Member)

Tanzim Reza
Lecturer
Department of Computer Science and Engineering
School of Data and Science
Brac University

Thesis Coordinator:
(Member)

Md. Golam Rabiul Alam
Associate Professor
Department of Computer Science and Engineering
School of Data and Science
Brac University

Head of Department:
(Chair)

Sadia Hamid Kazi, PhD
Chairperson and Associate Professor
Department of Computer Science and Engineering
School of Data and Science
Brac University

Abstract

Due to our unhealthy diets and the consumption of enhanced cholesterol in our daily lives, our health has become vulnerable and at risk of different types of cardiac diseases. The most common of them is Myocardial Infarction (MI), also known as Heart Attack. Myocardial Infarction takes place because of sudden blockage of blood flow in one's heart. Without sufficient blood flow, one's heart muscles cannot get the nourishment and oxygen that they need to function appropriately, which causes irreversible damage to the heart tissues. However, early detection and treatment of a Myocardial infarction can decrease the risk of heart damage and increase the rate of survival. As a diagnostic tool, the Electrocardiogram (ECG) is one of the most popular to diagnose various cardiovascular illnesses, including Myocardial Infarction (MI). The ECG captures the heart's electrical activity and these signals can be utilized to diagnose irregular cardiac rhythms. Because of the intensity and duration of ECG signals, manual ECG signal diagnosis is prone to errors and is neither sensitive nor specific for MI diagnosis when used alone. Therefore, this research proposes a novel approach of detecting Myocardial Infarction (MI), using deep learning techniques. It includes ConvNet model as well as other popular transfer learning models like MobileNet, VGG16 and InceptionV3 which uses 12-lead ECG signals as input. The trained model with the proposed ConvNet and MobileNet architecture have shown exceptionally promising accuracy in MI detection compared to VGG16 and InceptionV3. The performance of the proposed models are measured using Confusion matrix , Precision score, F1-score, Recall score and ROC curve. Our average accuracy is 97.50 percent which is acquired by using MobileNet. Also, the Convnet model shows promising result. Thereby, we can say that the suggested model can deliver high MI detection performance in wearable technologies and intensive care units.

Keywords: Myocardial Infarction, Deep Learning, ECG signal, CNN, Transfer Learning, ConvNet, VGG16, MobileNet, InceptionV3.

Acknowledgement

Firstly, all praises goes to the Almighty Allah who has given us the opportunity and strength to complete our thesis journey without any major hardship.

Secondly, We would like to convey our humblest gratitude to our thesis supervisor Dr. Jia Uddin sir, an amazing persona who constantly supported and motivated us throughout our research journey. He always encourage us to think out of the box so that we could approach things dynamically and learn more deeply. His kind cooperation along with his vision, knowledge and passion for his field of work has deeply inspired us throughout the thesis journey. We feel much privileged to have Sir as our supervisor.

Lastly, we are conveying our thanks to the Computer Science and Engineering department of BRAC University for providing us the opportunity of conducting this research and for giving us the chance to complete our Bachelor's degree and investing much finance and efforts for our research and publications.

Table of Contents

List of Figures

List of Tables

Nomenclature

The next list describes several symbols & abbreviation that will be later used within the body of the document

AV AtrioVentricular

CABG Coronary Artery Bypass Graft

CDC Centers for Disease Control and Prevention

CHF Congestive HeartFailure

CNN Convolutional Neural Network

ConvNet Convolutional Network

CVDs Cardiovascular Diseases

ECG Electrocardiogram

FibT Fibrinolytic Therapy

KNN k-Nearest Neighbors

MFCC Mel Frequency Cepstrum Coefficient

MI Myocardial Infarction

PCI Per-cutaneous Coronary Intervention

ReLU Rectified Linear Activation Unit

SA SinoAtrial

SSD Single Shoot Detection

SVM Support Vector Machine

Chapter 1

Introduction

Myocardial Infarction (MI), is also known as heart attack. The term ”Myo” refers to muscle, and ”Cardiac” means the heart, while ”infarction” refers to tissue death caused by lack of blood flow. Globally, Myocardial infarction (MI) is one of the leading causes of mortality in patients with cardiovascular diseases (CVDs) [38]. MI kills over 9 million people per year, according to the World Health Organization (WHO), and this figure is anticipated to climb to more than 12 million by 2030 [38][29].

A myocardial infarction happens when one's coronary arteries get blocked, leading to insufficient blood flow in one's heart. It hampers the functionality of the heart. Due to this, the heart tissues get damaged. According to Harvard Medical Schools' health publication, Blood clotting (thrombus) is the most common cause of a sudden blockage in a coronary artery. A blood clot usually originates inside a coronary artery that has already been narrowed by atherosclerosis. It is a condition in which fatty deposits (plaques) build up along the inside walls of blood vessels, narrowing them. MI is divided into three severity levels based on the time after the beginning of symptoms: early MI (EMI), acute MI (AMI), and chronic MI (CMI). Early diagnosis of these phases in clinical practice is crucial for implementing suitable life-saving treatment techniques such as fibrinolytic therapy (FibT), percutaneous coronary intervention (PCI), and coronary artery bypass graft (CABG) [29].

Fibrinolytic therapy (FibT) can save the entire myocardium at risk by the reperfusion of the blocked artery within 40 minutes [6]. This stage of MI can be reversed, which is known as EMI [6]. Cell death begins after 12 to 24 hours of blockage, resulting in irreparable myocardial damage. This stage of MI is known as AMI, and reperfusion with PCI or CABG can help with partial myocardial salvage [6]. If the blockage is untreated for more than 24 hours, the myocardium at risk becomes necrotic [6]. CMI refers to an irreversible condition of risk. It induces significant ventricular remodeling and also has a ninty percent death rate [5].

The Electrocardiogram (ECG) is a helpful diagnostic tool for a variety of cardiovascular diseases (CVDs) like myocardial Infarction (MI) or heart attack. ECG is used to evaluate the electrical and muscular activity of the heart. Electrodes and leads are used to measure electrical signals. An electrode is a conductive pad that is affixed to the human skin and it allows the electrical currents to be recorded. It records the electrical changes that occur during depolarization and repolarization during each cardiac cycle [28]. Electrodes are attached to the patient's limbs as well as the chest surface. The magnitude of the heart's electrical potential is then measured from twelve different angles, referred to as leads. The bipolar limb leads (I, II, III) are put in two different positions on the body. In contrast, the unipolar limb leads (aVR, aVL, aVF) are positioned at a virtual reference point with a zero electrical potential located in the center of the heart. The precordial leads V1 to V6 are the set of chest leads. Usually, an ECG evaluation includes determining the rate and analyzing the rhythm, both of which are predicted by the PQRS wave. P wave observes the atrium. The PR interval is the time it takes for electricity to move from the SA node to the AV node. The ventricle is examined using the QRS complex. The QRS interval is a measurement of electricity flow through the ventricles. The T wave assesses the recovery stage of the ventricles as they refill with blood. The QT interval is a measurement of how long it takes for the ventricles to heal and prepare to beat again. These characteristics are widely used to distinguish between normal and abnormal cardiac activity [28].

Deep learning is a sub-sector of machine learning in artificial intelligence that uses neural networks to learn unsupervised, unstructured, or unlabeled data. Deep neural learning or deep neural network are other terms for the same thing. Before Deep learning, there were Machine Learning Based models. Those ML based frameworks consist of manual feature extraction, selection, and classification. In ML-based models, Threshold based, K-nearest neighbor (KNN), support vector machines (SVM), random forest, and neural network (NN) are some of the most often used classifiers [29]. On smaller datasets, these techniques offer good results, but they have some substantial limitations. To begin with, quantifying progressive MI changes using fixed handmade characteristics is tough and feature extraction is dependent on medical expertise [29]. Secondly, the robustness of fiducial point detection is critical to the performance of time base methods, and false MI alarms are standard in the presence of various noises [9]. To overcome these constraints, the DL based models are used to detect MI from ECGs. Unlike ML approaches, DL models do not require manual feature extraction since they automatically identify high-level feature representations which are needed for effective diagnosis from raw data [24]. Studies across a variety of areas reveal that DL models outperform ML approaches in terms of generalization[24]. Furthermore, Transfer Learning Architecture is a more advanced version of Deep Learning Architecture in which models trained on one problem are used as a starting point for a related problem. Transfer learning is adaptable, allows pretrained models to be used directly for feature extraction preprocessing and incorporated into whole new models. Keras enables users to access several ImageNet image recognition models, such as VGG, Inception and ResNet [48].

1.1 Research Problem

According to the Centers for Disease Control and Prevention (CDC), every 40 seconds, a heart attack occurs but every 1 in 5 cases, a silent attack. That means the victim is unconscious of the fact that he is having a heart attack [30]. Approximately 15.9 million MI cases occurred worldwide in 2015 [8].

Sometimes MI occurs due to a severe spasm in the coronary artery (uncontrollable tightening muscle). It is believed that the leading cause of severe spasm in the coronary artery is smoking, emotional stress and pain, extreme cold and drugs [40]. A solid number of MI cases are asymptomatic or have minor symptoms. This kind of MI is known as silent myocardial Infarction. These cases are primarily found during regular checkups that reveal the presence of abnormal Q-wave in ECG reports. People with hypertension and with a history of cardiovascular disease (diabetes) are primarily at risk of silent myocardial Infarction [3].

Sudden Cardiac arrest can happen at any time without any alarm. A heart attack increases the risk of cardiac arrest. Arrhythmia (Irregular heartbeats) caused by electrical malfunction usually triggers cardiac arrest. When cardiac arrest occurs, the pumping action of the heart gets interrupted, and the heart cannot pump blood to the brain, lung, and other different organs. If the victim does not receive any treatment in time, death is inevitable. Heart attack is mostly responsible for cardiac arrest. However, cardiomyopathy, heart failure, arrhythmias, ventricular fibrillation, and long Q-T syndrome can also be the reasons [35].

About one-half of acute myocardial infarction patients suffer from Congestive Heart Failure (CHF). It is defined as a clinical syndrome by the moist rales (cavities in the lungs) at the lung bases continuing after vigorous cough, accompanied by a third heart sound, tachycardia, and tachypnea [1]. CHF is a pathophysiological illness in which the patient has improper filling or emptying of the left heart chamber (or both). It is considered the final stage of diverse heart diseases. CHF cases are increasing day by day. About 26 million adults were treated with CHF in 2014. Most CHF patients lose their quality of life because of debilitating symptoms like shortness of breath and fatigue. Also, most CHF-affected patients are elderly people (age > 64) [19]. The mortality rate of CHF patients is nearly three times higher than the of patients with MI [1].

Cardiogenic Shock (CS) is one of the main reasons for the death of patients with acute MI. CS occurs when the heart cannot pump oxygen and blood to the brain, kidney, and organs. Like Cardiac arrest, CS is also considered a medical emergency that must be diagnosed as quickly as possible. MI damages the heart muscles and tissues that lead to cardiogenic shock. However, heart failure, medicine side effects, chest injury, blood clots in the lung can also trigger cardiogenic shock. CS can occur in two ways. When MI interrupts the blood supply, the muscles that anchor the heart valves stop working. As a result, blood can not flow between heart chambers. Due to Infarction caused by MI, the heart can not pump blood properly and that causes CS. Another way is, MI damage can break down the wall that separates the ventricles. Without the wall between ventricles, the heart cannot pump blood, which also causes CS. Again, heart failure and arrhythmia can hamper the heart's ability to pump oxygen-rich blood to different vital organs [37].

Ventricular Aneurysm is a dangerous outcome that can happen after acute myocardial Infarction. Approximately 1.5 million people in the US develop myocardial Infarction. About thirty to thirty five percent of them suffer from left ventricular aneurysms. There are two major risk factors for developing a left ventricular aneurysm. The first one is total occlusion of the left anterior descending artery, and the other is failure to achieve patency of infarct site artery [34]. An Aneurysm is basically a bulge in the wall of an artery. If the aneurysm bursts, it can cause fatal internal bleeding [41]. Most ventricular aneurysms are asymptomatic and can be cured by routine treatment. However, left ventricular aneurysm symptoms with thromboembolic, arrhythmic, wall motion abnormalities, ventricular tachyarrhythmias can increase the risk of sudden death [41]. So, diagnosing MI and heart rhythm disorder at an early stage is very important to prevent further risk factors in life. For this, an ECG test is necessary for certain people who could have a risk of heart attack [23].

The Electrocardiogram (ECG) is the primary process to diagnose any heart condition and is considered the main diagnostic tool for cardiovascular diseases [25]. Even though ECG is a lifesaving tool, ECG signals different kinds of noises: baseline wander, powerline interference, electrode motion artifact noise, Electromyographic (EMG) noise [22]. The actual measured result can be badly hampered by the presence of noises like muscle noise and lung noise [2]. Baseline frequency is a noise with a low frequency around 0.5 to 0.6 Hz, which is transmitted by breathing, losing sensor contact, and body movement [22] [2]. Baseline noise primarily affects the low-frequency portion of ECG waves [2]. Sometimes, power supply noise at 50Hz frequency interrupts the ECG signal. This noise is described as powerline interference [23].

Thus, Myocardial Infarction is responsible for causing different fatal diseases. So, quick treatment to cure MI is essential. By classifying 12-lead signals from Electrocardiogram (ECG), experts can identify the location of MI in the body. However, different noises can be the reason for false results. But in deep learning CNN algorithms, there is a high accuracy to detect MI conditions by classifying 12-lead ECG data even with the presence of noise [20].

1.2 Research Objectives

The goal of this research is to detect Myocardial Infarction (MI) by classifying the 12-Lead ECG signals using the Deep Learning methods. Convolutional Neural Network (CNN) is a Deep Learning Algorithm that works from end to end that can classify an image and recognize it with high accuracy. This research will be conducted by the CNN method. The objectives of this research are:

- To deeply understand 12-lead ECG Signals.
- To deeply understand the CNN Algorithm and its efficiency.

- To develop a method where the CNN Algorithm receives ECG signals as input and comes up with the desired output after classifying them.
- To evaluate the proposed model.
- To offer recommendations on improving the model.

Chapter 2

Literature review

A significant amount of research has been conducted around the world in order to diagnose cardiac disease more quickly and effectively. The Electrocardiogram (ECG) is a common primary step in diagnosing cardiac disease. Researchers mainly focus on ECG image processing and analyzing to identify and classify different heart diseases. For this purpose, different algorithms are used separately or step by step, and different models are being upheld.

2.1 Electrocardiogram (ECG)

The Electrocardiogram (ECG) is a noninvasive diagnostic tool that can detect visible abnormalities in cardiac diseases. Manual ECG signal diagnosis, on the other hand, is prone to mistakes due to the small amplitude and time duration of ECG signals and is neither sensitive nor specific for CHF diagnosis when used alone. ECG signals diagnostic objectivity and dependability may be improved by using an automated computer aided system [19]. The electrical activity of the heart, while it operates, is represented by an electrocardiogram (ECG). For each cardiac cycle, the ECG waveform has some essential elements through which the activity of the heart is evaluated. P and T waves, PR and ST segments, PR and QT intervals, and the QRS complex are all characteristics of typical electrocardiogram (ECG) waveforms. ECG is used to convey information about various cardiac illnesses that a person may have in order to ensure efficient treatment. A typical ECG wave was taken from the article [44] which is shown in the Figure 2.1.

The precise points in (Figure-1) identified in the trace of an ECG are denoted with the letters P, Q, R, S, and T, according to international norms, and include the following: The P wave is the first wave in the ECG cycle, and this is a slight deflection that signals atrial depolarization, often known as ”atrial contraction”. T wave signifies ventricular depolarization, often known as ”ventricular relaxation”. Q, R, and S waves: these waves combine to produce the QRS complex. The contraction of the ventricles, or depolarization complex Electronics of the ventricles, is represented by

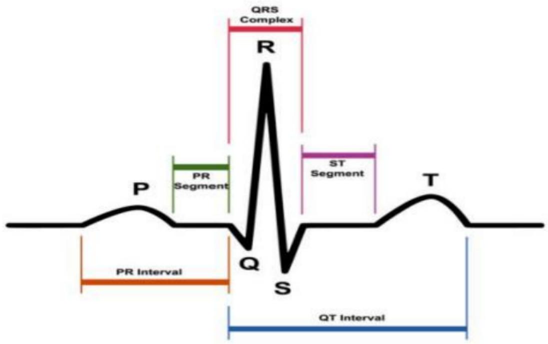

Figure 2.1: A typical electrocardiogram (ECG) waveform and its characteristic patterns (P and T waves, PR and ST segments, PR and QT intervals, as well as the QRS complex).

the QRS complex. The depolarization of the inter-ventricular septum is represented by the Q wave, the depolarization of the significant mass of the ventricles is represented by the R wave, and the ultimate depolarization of the ventricles at the base of the heart is represented by the S wave. The PQRST complex is made up of the P, Q, R, S, and T waves when added together. The ”R-R interval,” which corresponds to a cardiac cycle, is a term used by cardiologists to describe the interval between two PQRST complexes [26]. The PR interval, also known as the PQ interval, is a stretch created by the P wave and the PR segment (rectilinear stretch) that starts with the P wave and ends with the QRS complex. This interval is the time it takes for the depolarization wave to propagate from the atrial sinus node along the portion of the heart's electrical conduction system [26]. When an ECG is done on a patient with cardiac disease, the waveform differs from that depicted in the figure. For example, the QT interval may be more prolonged than usual, suggesting ventricular arrhythmia; the ST segment may be elevated, suggesting myocardial infarction [4].

2.2 Related Works

The research work [20] proposes an idea to detect MI and its location using the deep learning 10-layer CNN method for 12-Lead ECG signal. The 12-Lead data from ECG signals can be optimized and used as an input for the treatment of MI. In the field of machine learning, linear or non-linear features can be extracted from ECG signals using different kinds of extraction methods. These features are handcrafted and used as an input to classify ECG signal classification. However, deep learning algorithms can gather all these processes into a single structure that automatically extracts the hidden signatures from the raw data and uses this information to classify ECG signals. Convolutional Neural Network (CNN) is an end-to-end model that successfully classifies 12-lead ECG signals with high accuracy and sensitivity performance. Also, the CNN model doesn't require handcrafted features for extraction. Based on the presence of MI ECG stress in various contiguous ECG vectors, ten probable MI locations can be identified on a 12-lead ECG: anterior (A), anterior lateral (AL), anterior septal (AS), inferior (I), inferior lateral (IL), inferior posterior (IP), inferior posterior lateral (IPL), lateral (L), posterior (P), and posterior lateral (PL). Before sending data to CNN as input, the first ECG signals for the 12-leads are digitized at a sampling rate of 1000HZ. After that, a preprocessing method based on wavelet transformation is requested in order to reduce noise and baseline wander. These preprocessed data are segmented in order to detect R-peak. The CNN model will show ten numerical values for each 12-lead signal that will determine the probable MI locations. Usually, the Lead 2 signal is used to localize MI in the body. In this search paper, a 10-layer CNN has been used. A total of 651 sample signals are sent as input in the input layer. Then they move through the hierarchy ordered convolution layer and max-pooling layer. There they transformed into a featured map, and the dense layer uses this featured map to predict the class. For the purpose of evaluation, the data from ECG from all 12-leads is divided into seventy percent training, fifteen percent validation and fifteen percent testing. This 10-layer based CNN method may provide accurate results, but the method is impractical for portable devices. Convolution layers with a large number of depths or kernels in CNN, will require more computation cost.

The research work [26] focuses on distinguishing standard ECG signals and two other kinds of ECG signals with heart diseases caused by MI. using the 5-layer CNN model. The ways to classify ECG signals include the number of features, feature names, preprocessing techniques, database, modeling techniques, performance measures and accuracy. The CNN-based classification network converts the ECG signal from one domain to another, such as wavelet, frequency domain, Mel-Frequency Cepstral Coefficient (MFCC), and so on. The performance of this classifier can be divided into the following three classes: normal, atrial premature beat and premature ventricular contraction. When an ECG is performed on an infected person, the ECG will not be the same as a normal person's ECG graph, for example, the ST segment may be elevated, indicating MI.MIT-BIH is a database that contains 47 subjects and it is a large collection of recorded physiological signals. The database is based on three classes. This CNN method is mainly the mixer of 1-D convolution layer, Batch normalization layer, ReLU layer (non-linear unit), Softmax. CNN's is a special kind of network that processes data known as grid-like topology like time series data which may be considered as 1-D grid or convolution layer that takes samples at regular time intervals. After each convolution, batch normalization is performed in order to avoid explosion of the parameters. The ReLU layer decreases the problem of data overfitting. After the process, ECG signals are sent to the CNN network as input for classifying normal, atrial premature beat and premature ventricular contraction. The 5-layer CNN is a method with an accuracy rate. However, the method is not designed to figure out the location of MI in the body. Also, it can not recognize multiple heart diseases.

In this study, the author [29], created a novel multi-lead diagnostic attention-based recurrent neural network (MLDA-RNN) for automated diagnosis of the three MI severity phases in HC patients. The approach methodically examines the 12-lead ECGs for best categorization, capturing the multi-scale temporal relationships from each ECG lead. RNNs are used to first encode the temporal variations in 12-lead ECG signals. The intra-lead attention module summarizes the within-lead discriminative vectors and generates lead-attentive representations using these encoded vectors. Based on their clinical importance, the inter-lead attention module collects these representative vectors into a high-level feature representation. Using 12-lead ECGs from the PTBDB and STAFF III datasets, the author's proposed model obtained an overall accuracy of 97.79 percent without affecting class-wise recognition rates [29]. The quality-aware block, the RNN encoding block, the intra-lead attention module, the inter-lead attention module, and the classification block are the five processing blocks suggested in the research [29]. As learned attention weights often coincide with physicians' methods of recognizing MI severity phases, the MLDA-RNN presents promising findings for model inter prediction with improved performance [29].

According to [28], electrical activity provides subsidiary information about one heart's functionality through a real-time signal. Computer-aided technologies must be used to efficiently and precisely diagnose MI applying Artificial Intelligence. For the research, a multilayer deep Convolutional neural network (CNN) structure is presented to predict MI. Even with noise, the author [14] analyzed the 12 lead ECG data to achieve accurate automated MI identification. The novelty of the model presented in the research has a high classification accuracy for myocardial infarction prediction. The results are compared to models that include data augmentation and models that do not include data augmentation. The dataset for this study was gathered from the PTB Diagnostic ECG database, which has open access to all [28]. In this article [28], Each signal record is made up of 15 quantifiable signals. The 12 leads are unipolar (i, ii, iii), augmented limb leads (avr, avl, avf), precordial leads (v1, v2, v3, v4, v5, v6), also known as conventional leads and Frank leads (vx, vy, vz). Deep convolutional NN layers are used in the suggested system. The first convolutional layer is the input layer, which has a filter size of 4 or 8 for detecting edges and a kernel size of 33 with a 128 input shape and a ReLU activation function. The other convolutional layers are identical to the first, except for the filter size, which has been changed. In the pooling layer, the maximum pooling operation is chosen and the value for the dropout layer is determined at random. The fully connected layer with respective dense units and activation function is the output layer [28].

To improve the training dataset, data is processed and enhanced with specific image augmentation techniques. For performance enhancements, image augmentation of data is used. Image augmentation techniques offer a variant of transformations that may be used to produce a new data set from existing data. Over fitting can be considerably decreased by employing these transformation strategies. Flipping, color space, cropping, rotation, translation, noise injection, gamma correction, brightness, blurring, and other operations are among the transformation techniques. However, not every transformation operation can be applied to this data set. Operations such as noise injection, gamma correction, brightness and blurring can only be applied [28]. For training and testing, the data is partitioned using varied splitting ratios (with major portions such as 80.01 percent or 20.00 percent for training and either 20 or 30 percent for testing). Performance metrics such as confusion matrix and accuracy are evaluated using the trained model. The proposed CNN model with data augmentation was found to have the best classification performance, with a score of 94.98 percent, compared to the proposed CNN model without data augmentation, which scored 90.34 percent, and the alternative CNN model of ECG beats with noise, which scored 93.53 percent [28]. As a result, the proposed model in the research has high classification accuracy for myocardial infarction prediction.

The research [17] proposes a 1D Convolutional Neural network model with a 16-layer deep network structure for automatically classifying 1000 fragments of 10-s ECG signal recognizing 17 class data set. In this model, the researchers found around 91.33 percent accuracy level without any signal filtering and use of QRS complex detection. In this model, 3600 samples of long-duration raw ECG signals are classified, and this model only required 0.015 seconds to classify each sample. Moreover, as it is fast and reliable, it can be used in mobile devices and cloud computing. However, the accuracy can be improved a lot if they used a 2D CNN model for their dataset. The research [18] proposes both 1D and 2D CNN models and derives a comparison between both models' performance measurement. Here, both types of deep CNN are trained with ECG data. In the 1D model they put ECG data as input in the network, and for the 2D model they used Short Time Fourier Transform. Here, in the 2D model, they use 6 convolutional or hidden layer networks. Furthermore, they explained how deep learning performs far better than machine learning algorithms, as in deep learning, there is no need for manual feature extraction and data preprocessing steps. In deep learning, network features are extracted and analyzed automatically. Although this model performed quite impressively compared to the previous one, it is comparatively slow without using a GPU.

The research [32] talks about Single Shoot Detection (SSD) Mobile Net v2-based Deep Neural Network architecture and how with around ninety eight percent accuracy MI and some other cardiac diseases can be detected with the model. They used 12 lead-based ECG image processing and for the data set, they manually collected ECG images from the cardiac institute and had to do lots of data preprocessing because online time series-based data couldn't suit their model.

According to research [13], ECG actually helps to diagnose most heart problems. It makes diagnosing AV block, myocardial Infarction, and ventricular tachycardia easier. Almost hundreds of millions of ECGs are done every year. At the moment, various machine learning (ML) techniques are available for categorising ECGs. However, there are certain drawbacks, such as the usage of heuristic manual labour or engineered features in conjunction with a shallow feature learning architecture. To overcome it and make the ECG more accurate, a deep learning architecture is proposed where the first layer of convolutional neural behaves as extraction and final decision is done by FNC layers. ECG classification is essential and has strong attention not only for medical purposes but also for computer scientists, especially in the AI field. In the most recent results, the scientists used a 4-layer convolutional neural network and outperformed the average cardiologist in both recall and precision. Database has almost 500 times the amount of unique patients than competitors. Other efforts include two-class ECG classification algorithms with minimal neural architecture that perform well. However, for a variety of reasons, most researchers are unable to use this. Data augmentation approaches are used as a solution. Data and problem statement: The current classification ECG determines which class patients can be classified.

There are 4 classes of ECG
(1) normal sinus rhythm
(2) erratic
(3) other kinds of rhythm
(4) boisterous

Considering ECG data obtained from patients decides which machine learning algorithm should be used. To begin, the majority of ECG data is time-series data with a period of 30-60 seconds and a sample rate of around 0.003s. There are several machine learning methods that deal with series data. Feature engineering and automated feature extraction provide an artificial intelligence solution. There are just a few figures that demonstrate how deep learning architecture works. It may be utilized in ECG classification when the training data is unstructured and imbalanced and it can be represented as an ID time series with a conventional length for single-lead ECG.

Chapter 3

Methodology

The purpose of the proposed CNN-based MI detection model in Deep Learning as well as Transfer Learning is to detect Myocardial Infarction (MI) from analyzing 12-lead ECG signals. After the Preprocessing stage, For better accuracy in research it normalizes the ECG signal images according to the input requirements of the proposed models. The ECG images with different sizes suitable for the model are taken as input and split into train, validation, and test sections and passed to the CNN model. The CNN model has mainly three types of layers and they are Convolutional layer, Pooling layer, and Fully Connected layer. The convolutional layer is used to produce tensors with applying filters, and the max-pooling layer does the subsampling of images and reduces the size of the image. After passing through a number of convolutional and max-pooling layers, the data is flattened and passed through a compressed fully connected neural network for quick and smooth classification of MI affected class, normal class, history class and Abnormal class based on the ECG images. Figure 3.1 shows a flow chart view of the model which classifies two classes.

This MI detection process has five major stages, which are –

Input Data: At the beginning a dataset containing different classes of ECG images has been collected.

Data Preprocessing: Different ConvNet architectures take input with different sizes. So, this method is applied so that the input image can meet the requirements of the models.

Models: Two CNN models have been built for classifying both two classes and four classes of ECG images. Finally Different Transfer Learning models have applied for fine tuning.

Result Analysis: The performance of the models are judged on this stage. Classification report. matrics and different graphs are used to analyze the accuracy and loss of the models.

Predictions: In this stage, images from different classes are used to test how well

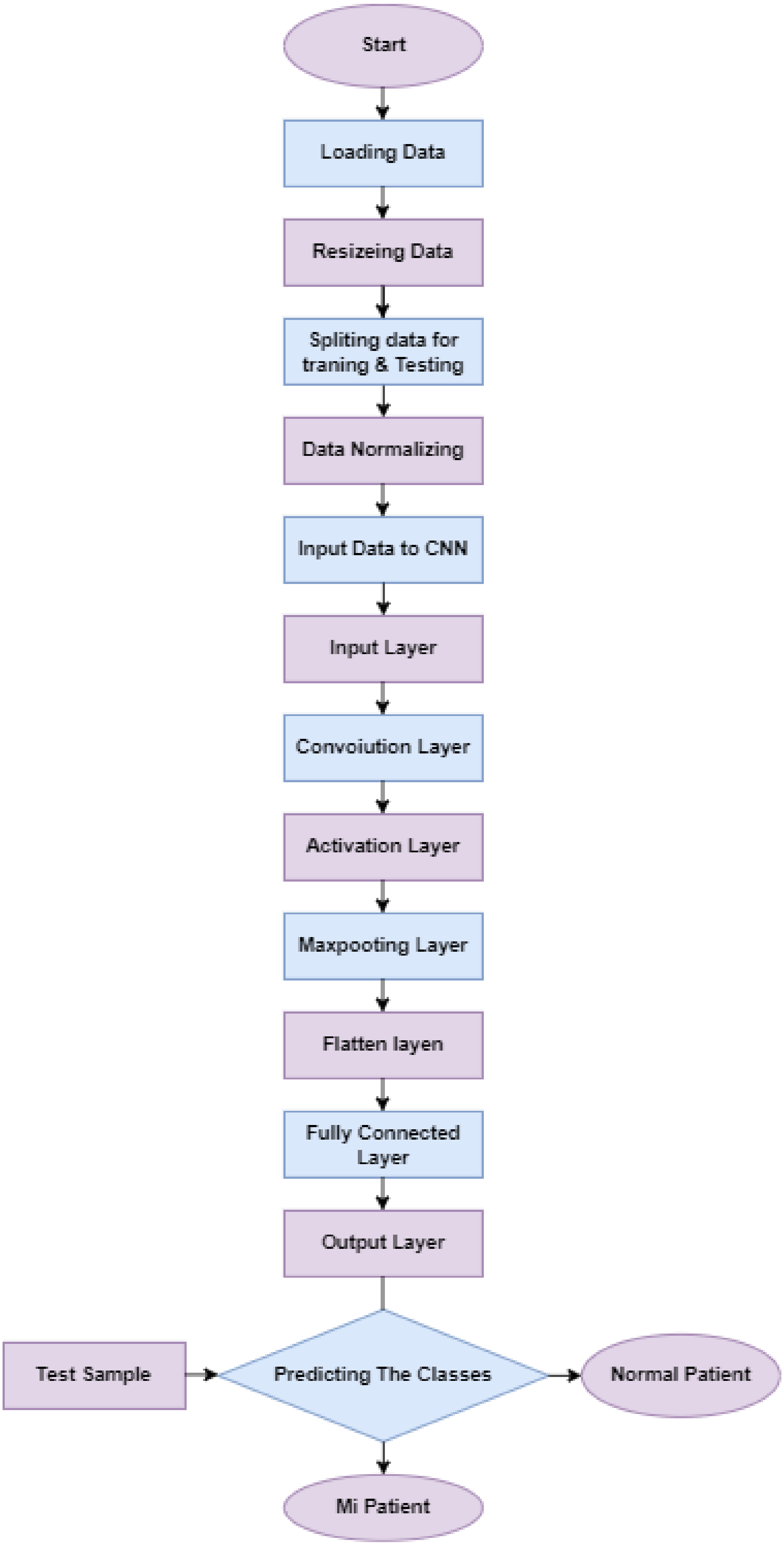

Figure 3.1: Workflow of for two class

the proposed models are differentiating their classes.

Figure 3.2 represents the workflow proposed MI detection model for four classes.

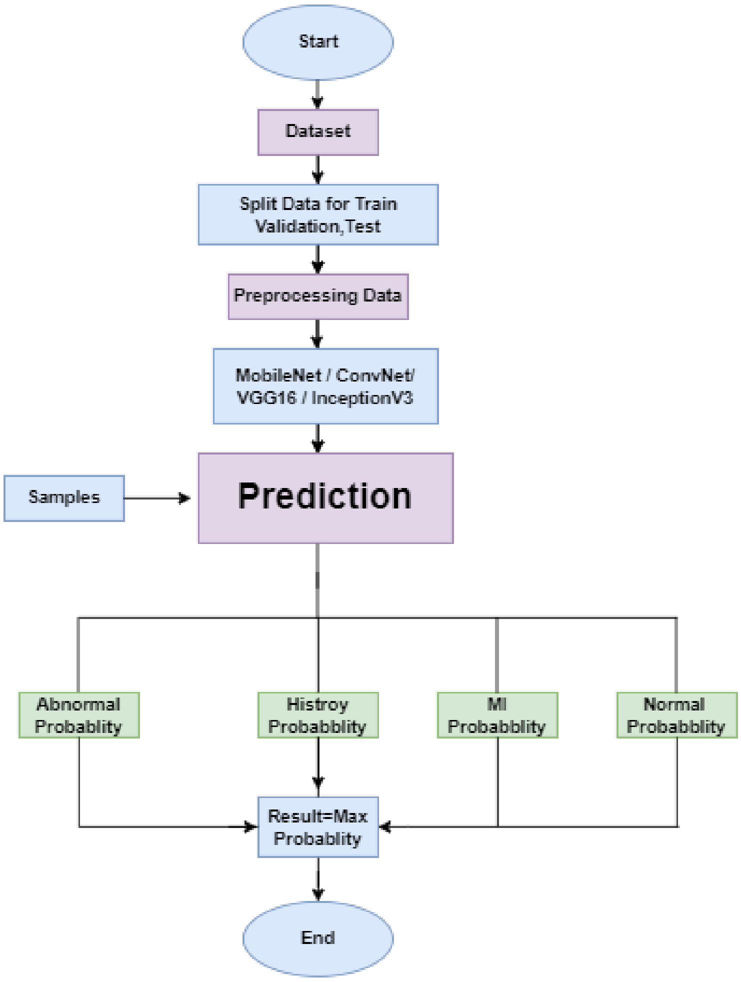

Figure 3.2: Workflow of for four classes

3.1 Dataset

When it comes to input data for myocardial Infarction (MI) detection there are very few image bases dataset available and for this analysis raw image data is something which was preferred as CNN algorithm works better with images also VGG16, MobileNet are pre-trained model which are trained with imageNet. Therefore, obtaining the images of the ECG signal was a challenging part. On the other hand, most of the datasets that are available in open source are in CSV format and most of the research conducted in MI used those open-source datasets. However, after searching a dataset with raw images of ECG signal came to the notice and those are '.jpg' formatted data that are used for this research.

The dataset contained ECG images of Cardiac patients under the auspices of Ch. Pervaiz Elahi Institute of Cardiology Multan, Pakistan. This institute aimed to help in the future research on Cardiovascular diseases and to assist the scientific community in their journey of modern advancement. The dataset of cardiac patients was published on 19 March, 2021(version 2) by Mendeley Data which is contributed by Ali Haider Khan and Muzammil Hussain [31]. The raw dataset of images can be directly downloaded from this attached link below: https://md-datasets-cache-zipfiles-prod.s3.eu-west-1.amazonaws.com/gwbz3fsgp8-2.zip

Dataset contains 2213 height and 1572 width, 929 ECG RGB '.jpg' format images in four folders and the folders contain Abnormal, History, Mi and Normal classes images individually. Fig 3.3 shows the sample of grayscale ECG image. The 'Abnormal' folder has the 233 ECG images of the people who had abnormalities in their ECG report. The 'History' folder stores 172 ECG report images of the people who have experienced MI in the past. Normal 284 people's ECG report images are stored in the folder named 'Normal' and Myocardial Infarcted affected people's 240 images are stored in 'Mi' folder. 239 Mi class images and 240 images are used in the model that classifies two classes and 172 images from each class were collected for the models that differentiate four classes.

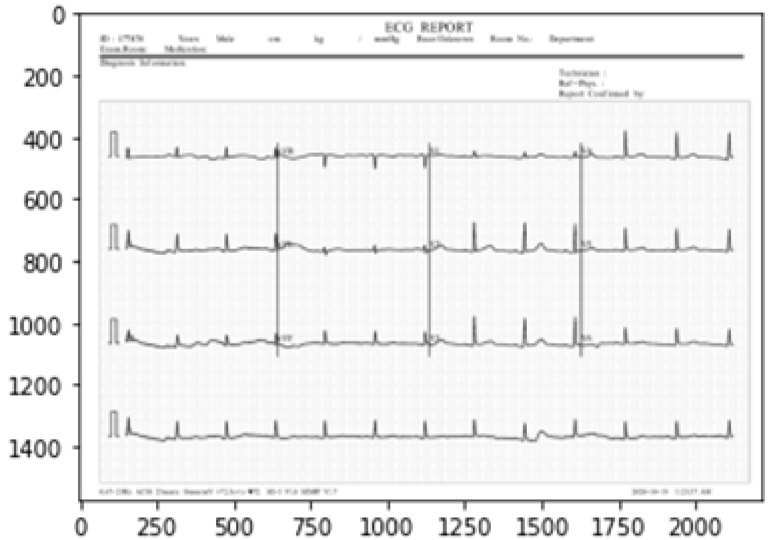

Figure 3.3: ECG signal Image of a MI patient

In the later stages of the research all four folders are used and the folder named as 'MI' has 239 images, 'Abnormal' folder has 233 images, so patients with abnormal ECG reports are stored in this folder. Likewise, The folder named 'History' has 172 images. These are the ECG reports of the people who experienced Myocardial Infarction. Normal people's 284 ECG reports are stored in the folder named 'Normal'. All the images are RGB images and in '.jpg' format. Patients with abnormal ECG reports are stored in this folder. 172 images from each of the classes have been selected for the research.

3.2 Data preprocessing

Data preprocessing methods are not the same for all models. The resolution of the images might not be the same and that causes an error during the training process. Again, a high pixel image requires higher computational power. Considering this, when a simple convNet model was used to identify Mi and Normal classes, the preprocessing strategy was different from others. In that case, (2213, 1572, 3) images were converted into (100, 100, 3) images so the six layer model can handle the complexity. However, raw data was resized into bigger size RGB images, when they were pushed into more powerful ConvNet and other Transfer Learning models. For two classes an average 80 percent images were collected for training and 20 percent for validation and for four classes among 688 images, average 80 percent data was selected for training, 17 percent was validation and 3 percent for testing. Fig 3.4 indicates the data preprocessing details.

Models	Resized Shape	Classes	Total Images	Quality	Training	Validation	Testing
ConvNet (6 layers)	100 x 100	2	479	Grayscaled	383	95	
ConvNet (13 layers)	200 x 200	4	673	RGB	548	120	20
MobileNet	224 x 224	4	673	RGB	548	120	20
VGG16	224 x 224	4	688	RGB	508	160	20
InceptionV3	224 x 224	4	682	RGB	548	129	20

Figure 3.4: Preprocessed image for different models

Fig 3.5 indicates the Resized and grey scaled input images in numeric formate.

```
                                          Category Label
0    [[255, 255, 255, 255, 255, 255, 255, 255, 255,...     0
1    [[255, 255, 255, 255, 255, 255, 255, 255, 255,...     1
2    [[255, 255, 255, 255, 255, 255, 255, 255, 255,...     0
3    [[255, 255, 255, 255, 255, 255, 255, 255, 255,...     1
4    [[255, 255, 255, 255, 255, 255, 255, 255, 255,...     0
..                                                 ...   ...
474  [[255, 255, 255, 255, 255, 255, 255, 255, 255,...     1
475  [[255, 255, 255, 255, 255, 255, 255, 255, 255,...     0
476  [[255, 255, 255, 255, 255, 255, 255, 255, 255,...     0
477  [[255, 255, 255, 255, 255, 255, 255, 255, 255,...     1
478  [[255, 255, 255, 255, 255, 255, 255, 255, 255,...     1

[479 rows x 2 columns]
```

Figure 3.5: Resized and grey scaled input images numeric form

3.3 Convolution Neural Network Architecture

Convolutional Neural Network (CNN) is a type of deep learning algorithm which we will use in research to identify MI. CNN is commonly used in order to analyze images and data as well as diseases [27].The concept of Deep Learning or Deep Neural Network refers to Artificial Neural Network with multiple layers. Deep Learning mainly deals with huge amounts of data. The ambition behind adding deeper hidden layers has begun to exceed classical machine learning methods performance in different fields especially in pattern recognition. Convolution Neural Network (CNN) is one of the most popular Neural Networks which consist of multiple convolution layers, non-linearity layer, pooling layer and fully connected layer. The name convolution comes from the mathematical linear operation between matrices called convolution. Convolution Neural Network has been showing groundbreaking results in various fields related to pattern recognition, from image processing to signal recognition [10].

In most Neural networks, the neurons or perceptions take some inputs then apply a dot product operation with weight and optionally follow it with non-linearity. At the end, it gives a score function and a loss function as an output. However, the Convolution Neural Network Architectures makes the direct assumption that the inputs are images which allows CNN architecture to have some certain properties that are suitable for image processing. Those properties make the forward function more effective and efficient to implement and decrease the number of parameters in the Neural Network [39].

Convolution Neural Network Architectures is known as ConvNet. ConvNet architecture is mainly a stack of layers that transforms an image input volume into an output volume. By far the most popular layers in Convolutional Neural Network are CONV, ReLU, POOL, and FC (Fully Connected). Each layer has certain properties and is distinct from the other. All these layers take an input 3D volume or 2D (without RGB) volume and transform it to an output 3D or 2D volume by passing through different differentiable functions inside layers. CONV and FC may contain parameters like weight and bias, but ReLU and Pooling may don't have any parameters. CONV, FC, and POOL layers sometimes contain Hyperparameters like L1 and L2 norm [39].

The CONV layer is the fundamental building block of the Convolution Neural Network that does the most complex computation. The first convolution layer takes the image as input. The convolution layers are made of a set of learnable filters, also known as features. CNN prevails these filters when the inputs dig into deeper layers. For example, in terms of image input, the filters at the first layer might be used to detect all the edges and the second layer filter can be used to detect different geometric shapes [10]. CNN slides each filter across the width and height of the input volume and calculates the dot product between the values of the filter and some certain location of the input. Since the filter slides all over the height and width of the input volume, the convolutional layer will produce a 2D array activation map that covers the responses of that particular filter at every location [39]. For example, [[0, 1, 0], [1, -4, 1], [0, 1, 0]], [[1, 0, -1], [0, 0, 0], [-1, 0, 1]] and [[-1, -1, -1], [-1, 8, -1], [-1, -1, -1]] these 3 x 3 filters are frequently used to detect the edges in the image [10]. While sliding across the input image, when a filter finds any edge, the dot product between the pixel values of that certain location of the image and the filter gives higher value in activation maps. Figure 3.6 shows the Convolution layer operation on an image.

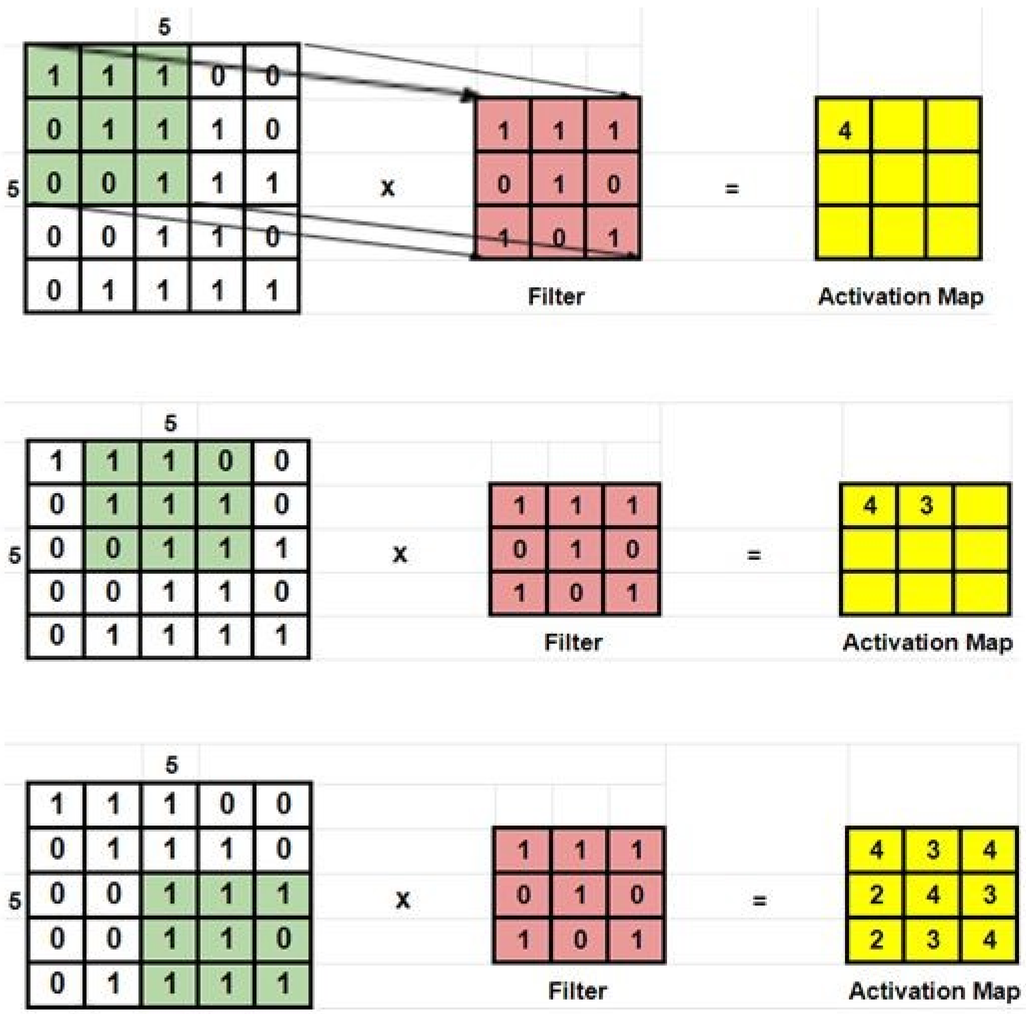

Figure 3.6: CONV layer operations

If the size of the activation map is N x N, size of the filter F x F and p, q represents the indexes of activation matrix O, the value at p, q index will be,

$$O_{p,q} = \sum_{i=1}^{F} \sum_{j=1}^{F} W_{ij}, X_{i+p-1,j+q-1} \tag{3.1}$$

Convolution layer contains some hyper parameters. Depth, stride and zero-padding, these three hyper parameters control the size of output volume. The depth hyper parameter defines the number of the filters that will be used in the convolution layer. Each filter will look for something unique in the input image. Assume, different neurons together at the depth dimension may activate because of the existence of edges and shapes at the input image [39]. Figure 3.7 shows an example of an input volume and an output volume [39].

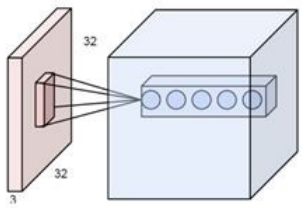

Figure 3.7: Example of an input volume and an output volume

Another hyperparameter is stride. Stride decides the amount of movement of the filter over the image. For example, if the value of stride is 1, the filter will slide one pixel at a time [43]. Figure 3.8 is an example of a one sliding process.

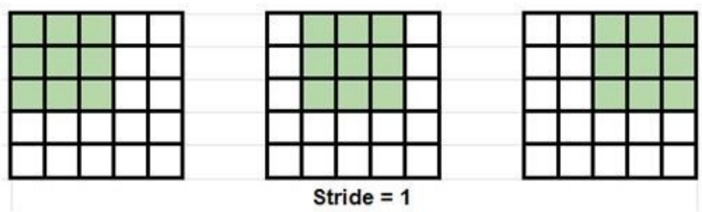

Figure 3.8: The filter sliding one stack of pixel at a time

A Convolution layer may lose information when the information exists at the border of an image. This information is only recorded when the filter slides over them. In order to solve this issue, zero-padding parameter has been used [10]. The number of zero-padding depends on the dimension size of the filter. If our filter size is F x F, the number padding can be achieved by using this equation[20].

$$NumberPadding = \frac{N-1}{2} \tag{3.2}$$

In the previous example, the size of the filter was 3 x 3. So, in the input image, there will be one amount of zero padding. Figure 3.9 represents zero padding.

0	0	0	0	0	0	0
0	1	1	1	0	0	0
0	0	1	1	1	0	0
0	0	0	1	1	1	0
0	0	0	1	1	0	0
0	0	1	1	1	1	0
0	0	0	0	0	0	0

Figure 3.9: Zero padding

The output volume and parameter sharing of the conv layer can be computed following some mathematical operations. Assume,

$$Inputvolumesize = W_1 * H_1 * D_1$$

$$Numberoffilter = K$$

$$Spatialextentorsizeofthefilter = F * F$$

$$Thestride = S$$

$$Theamountofzero - padding = P$$

The depth of each filter has to be same as the depth of input

$$TheOutputVolumeSize = W_2 * H_2 * D_2 \tag{3.3}$$

Where,

$$W_2 = \frac{W_1 - F + 2P}{S} + 1 \tag{3.4}$$

$$H_2 = \frac{H_1 - F + 2P}{S} + 1 \tag{3.5}$$

D2 has to be the same as the number of filters,So D2=k.

$$NumberOfParameters = ((F * F * D_1) + 1) * k \tag{3.6}$$

Next step after the Convolution layer is to apply non-linearity. The reason for applying the activation function is to introduce non-linearity to the convolution layer output. The term non-linearity means output cannot be produced from the linear combination of domains. In Deep learning, if the activation function is not applied, no matter how many layers are plugged, the neuron will act as if it is a neuron of only one layer [45]. Sigmoid function is a popular activation function. It is the mathematical form is

$$\sigma(x) = \frac{1}{1 + e^{-x}} \tag{3.7}$$

Sigmoid function squashes the domain into a range between 0 and 1. Figure 3.10 is a graphical visualization of Sigmoid [39].

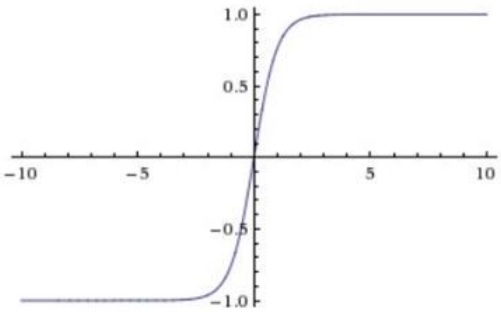

Figure 3.10: Graphical representation of Sigmoid

If the input value is high then the output of the sigmoid function will be 1 and if low the output will be 0. However, very high and low values can kill the gradient. Figure 3.11 shows the backpropagation for a sigmoid function [39].

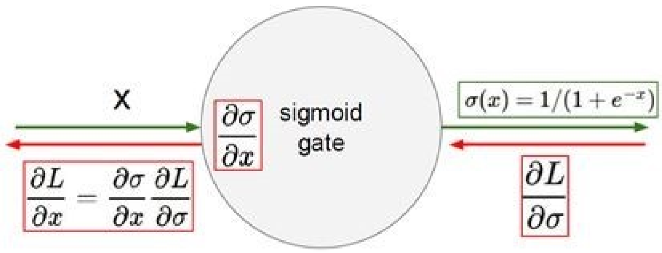

Figure 3.11: Back propagation of a sigmoid function

Assume, input (x) in a sigmoid function is -15 which is quite smaller number,

$$L = (x)$$
$$\frac{\delta}{\delta x}\sigma(x)$$
$$= \frac{\delta}{\delta x}\frac{1}{1+e^{-x}}$$
$$= \frac{e^{-x}}{(1+e^{-x})^2}$$
$$= \frac{1+e^{-x}-1}{1+e^{-x}} * \frac{1}{1+e^{-x}}$$
$$= (1-\sigma(x)) * \sigma(x)$$
$$= (1-0.000000305) * 0.000000305$$
$$= 0.000000305$$

So, when the input value is deficient, it is gradient looks near zero. From the sigmoid graph it is clear that the very negative region of the graph is flat. So the output is zero in those regions. When any upstream gradient comes down using chain rule, it basically becomes a multiply operation with zero. As a result, a minimal value flowers downward to the local gradient. In short, after chain rule sigmoid function kills the gradient flow and a zero gradient passes down to downstream nodes. Similar problems occur when input is very high. However, when input is 0, the backpropagation value becomes 0.25 which does not kill gradient flow. In conclusion when the input is near 0, the sigmoid function works effectively. Another problem of sigmoid function is, the exponential operation is costly [39].

ReLU or Rectified Linear Unit is another popular activation function for image domains. ReLU operates pixel wise. It replaces all negative pixel values with zero and keeps only the positive values [36]. ReLU has some advantages over sigmoid function. Like the sigmoid function, ReLU does not saturate the positive region which holds half of the input space. Also, ReLU converges much faster and more computationally efficiently than sigmoid [39]. ReLU function, mathematical represent as g(x) = max (0, x). Figure 3.12 is a graphical visualization of ReLU [36].

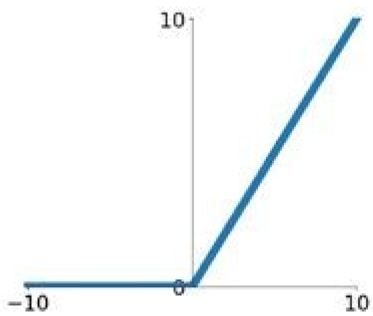

Figure 3.12: Graphical representation of ReLU

Pooling is an operation that is used to reduce the dimensionality of activation maps also known as feature map input while still preserving spatial invariance [36]. It also minimizes the amount of parameters and computations in the neural network. The pooling layer functions every depth layer of the feature map and resizes them using max operation. Most common pooling technique is called maxpooling2D. In most cases 2 x 2 applied with two strides down samples every depth slice in the feature map by two along both height and width in the pooling layer. So, every max operation takes the maximum value among four numbers. The depth dimension after pooling operation remains the same [39]. One stride also can be used for avoiding downsampling. However, this method is not very popular [10]. Like convolution layers, pooling layers output also can be computed by following some equations. Figure 3.13 shows Pooling operation on a 4 x 4 metrics [39].

Assume,

$$\begin{gathered} Inputvolumesize = W_1 * H_1 * D_1 \\ DimensionOfThePatch = F * F \\ Stride = S \\ OutputVolumeSize = W_2 * H_2 * D_1 \end{gathered} \tag{3.8}$$

Where,

$$W_2 = \frac{W_1 - F}{S} + 1 \tag{3.9}$$

$$H_2 = \frac{H_1 - F}{S} + 1 \tag{3.10}$$

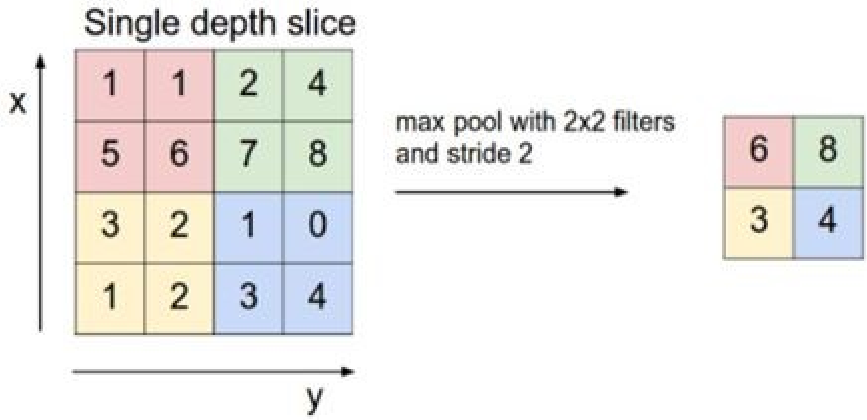

Figure 3.13: Pooling operation

The Flatten layer converts a multidimensional matrix to a single 1D array. The flatten layer is connected to the final layer of the convolutional neural network [39] [21]. The final layer of ConvNet architecture is the Fully connected layer also known as Dense layer. In the dense layer, all the neurons have a connection with all the activation in the previous layer. The activation can be calculated with matrix multiplication and summation of bias and applying an element-wise non-linear function [39]. In most cases fully connected layers are plugged at the end of CNN architecture. If x is the input of fully connected layer and W represents the matrix that contains the weight of connection between layers, then the output will be [15].

$$y = f(wTx + b) \tag{3.11}$$

After the ConvNet training process, a loss function is attached in order to establish the accuracy of the proposed model. Loss Function computes the distance between the result based on ConvNet prediction and the actual result. One of the most populer, Loss Function in Softmax. Softmax interpretes class scores specified by the model as unnormalized log probabilities and computes its cross entropy loss.

$$L_i = -log\left(\frac{e^{f_{yi}}}{\sum_j e^{f_j}}\right) \tag{3.12}$$

Where $e^{f_{yi}}$ is the exponent of the score of a particular class and $\sum_j e^{f_j}$ is the summation of all other classes exponent scores.[39]

Chapter 4

Implementation And Architecture

This part defines the implementation of 'Myocardial Infarction Diagnosis Using ECG Signals Applying Deep Learning Algorithm'. The proposed model was applied and tested in Google Colab. Colab is basically a python development environment that can be run in any browser. In this research Keras has been used. Keras is a high level deep learning API running on the Tensorflow which is a machine learning platform. It is a highly effective platform to solve machine learning problems focusing on modern deep learning. Keras can easily process huge amounts of complex data. Keras was developed targeting fast experimentation. It is user friendly and enables users to be more focused on parts of the problem rather than having a cognitive load. Keras and TensorFlow also reduces low level tensor flow operation on GPU and CPU.

CNN is a deep learning model that classifies higher-dimensional image data. Different libraries and their functions from Keras can be helpful for this image classification research.

4.1 Implementation

The proposed model can take both '.jpg' and '.png' format images as input. The complete model fetches image folder by folder and gives each folder's image a certain label to identify the image classes later. In this research 0 is used to label Abnormal patients and 1 is used to label patients with History, 2 is used for labeling Mi patients and 3 labels the Normal people class. Figure: shows the labeling per class. Python's OS, CV2 and Drive library has been used in resizing operations. Nested for loop and splitting function has been added in this program to split data for training, validation and testing. Preprocessing functions for different models are also used to resize according to their requirements. Figure 4.1 represents the labeling the class.

```
{'Abnormal': 0, 'History': 1, 'Mi': 2, 'Normal': 3}
```

Figure 4.1: Leveling the Classes

4.2 Input Data-prepressing

The dataset that has been used for this research contains '.jpg' format ECG signal images. The dataset has four image classes which are Abnormal, History, MI and Normal. For optimal results, the input data has to be preprocessed. Preprocessing can be accomplished by resizing the image to lower resolution, gray scaling and normalizing by dividing all pixel values by 255.0. The Dataset folder contains four folders.

4.3 CNN Algorithm Implementation for two classes

After pixel wise normalization, the CNN algorithm takes multi-dimensional lists as image input in the input layer. Since the images were reshaped, the dimension of an input image becomes 100 x 100 x1. In this research, 80 percent of images have been used for training and 20 percent has been used for testing. Figure 4.2 shows an image structure after normalization.

```
[[[0.00392157]
  [0.00392157]
  [0.00392157]
  ...
  [0.00392157]
  [0.00392157]
  [0.00392157]]

 ...

 [[0.00392157]
  [0.00392157]
  [0.00392157]
  ...
  [0.00392157]
  [0.00392157]
  [0.00392157]]]
```

Figure 4.2: An Image structure after Normalization

Next, the data enters the Convolutional layer. At the first convolution layer, 128 filters have been set and each filter is 3 x 3 x 1 dimensional. Usually at first convolution, filters are used for detecting edges of an image. CNN slides each of 128 filters across the width and height of an input image volume and calculates the dot product between the values of the filter and some certain location of the input image array. The movement for the filters depends on the value of stride. In this Research the default stride value one has been used. Also zero amount of zero-padding has been applied. The output volume and the number of parameters for 128 filters and for an image input can be calculated theoretically by plugging the Input dimension, filter dimension, number of stride and zero-padding values using equation (3).

Following the equations, the output volume size was theoretically 98 x 98 x 128 and the number of parameters was 1280. Figure 4.3 shows the activation map of an image after applying a convolution layer operation.

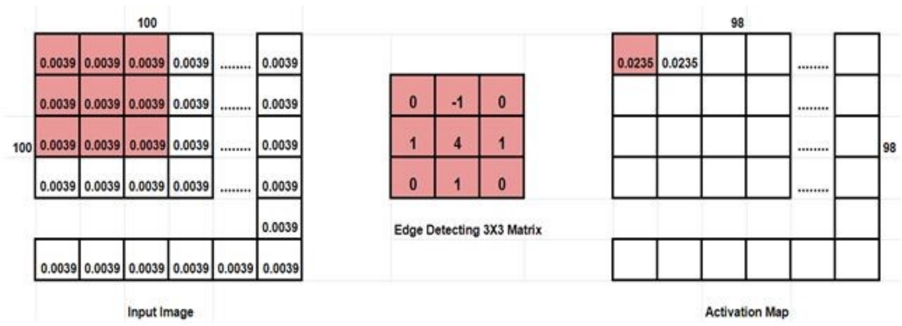

Figure 4.3: Activation map of an image after applying a convolution layer operation

The applied CNN algorithm also slows the same output size and number of parameters.

The output of the Convolutional layer is an activation map. This activation map then becomes the input for the Activation Layer. In this research both ReLU and Sigmoid activation function have been used. The relu activation function checks every pixel value and replaces all negative values with 0 and keeps only positive values. Similarly, the Sigmoid function squashes lower values to 0 and higher values to 1. Figure 4.4 presents how the ReLU function is applied on a pixel.

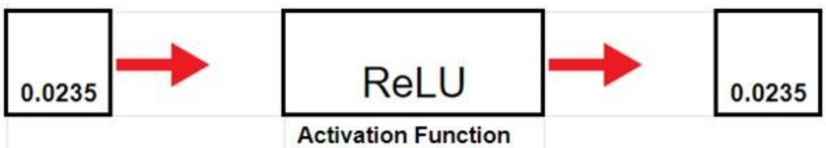

Figure 4.4: ReLU function applied on a pixel

Pooling layer takes place after the activation layer. This layer takes higher dimensional input and downscales it to lower dimension. Maxpooling2D, a common pooling function has been used in this experiment. Max pooling takes the maximum value between four numbers. The output volume of the maxpooling2D layer also can be computed. In this research, The Input in the pooling layer is 98 x 98 x 128 and patch dimension is 2 x 2. Therefore, the output will be 49 x 49 x 128 after applying equation (8). Figure 4.5 express Max pooling 2D operation

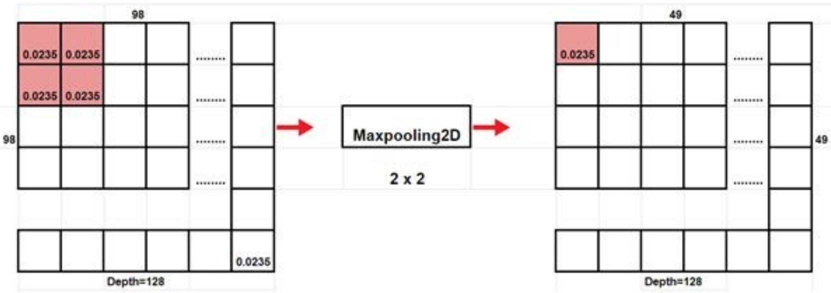

Figure 4.5: Maxpooling2D operation

Flatten layer converts the 49 x 49 x 128 Maxpooling2D output volume to a 1D list with 307328 elements. Figure 4.6 shows an 1D array at Flatten layer output.

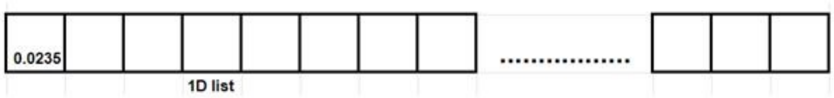

Figure 4.6: 1D array at Flatten layer output

Flatten layer output is basically the input for the following convolution layer. However, One Convolution layer with 128 filters, one ReLU activation function and one Sigmoid activation function, one maxpooling2D, dense layer and flatten layer have started to show high accuracy rate from average 15 epochs in this operation. So, adding more layers is unnecessary.Figure 4.7 is summary of proposed CNN model.

Fully connected layer is the last layer of a convolutional neural network. It connects outputs for each image operation then sends the result to the loss function to compute the loss amount for training and testing data. Figure 4.8 is a block representation of the proposed deep CNN model applied in this analysis.

```
Model: "sequential"
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
conv2d (Conv2D)              (None, 98, 98, 128)       1280
_________________________________________________________________
activation (Activation)      (None, 98, 98, 128)       0
_________________________________________________________________
max_pooling2d (MaxPooling2D) (None, 49, 49, 128)       0
_________________________________________________________________
flatten (Flatten)            (None, 307328)            0
_________________________________________________________________
dense (Dense)                (None, 1)                 307329
_________________________________________________________________
activation_1 (Activation)    (None, 1)                 0
=================================================================
Total params: 308,609
Trainable params: 308,609
Non-trainable params: 0
_________________________________________________________________
None
```

Figure 4.7: Summary of the proposed CNN Model

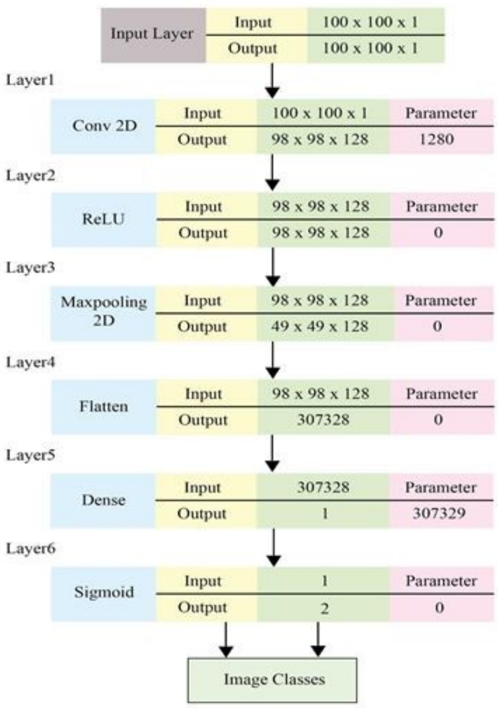

Figure 4.8: Block representation of the proposed deep CNN model

This research is conducted in multiple phases.In the beginning stage, the research was done in a smaller scale where 2 classes from the data set was used and a simpler CNN architecture was implemented to detect Myocardial Infarction. After getting promising accuracy with this Model the scale of the research was increased and implemented more depth CNN architecture to classify 4 classes which are ‘Normal’ , ‘Abnormal’ , ‘MI’ and ‘History’. Here each class has 172 ECG signal images.For example: In History class there are 172 images which show ECG reports of patients who have a previous record of MI.

4.4 13 layer ConvNet Implementation On four classes

In the depth CNN model, multiple layers are added in the purpose of increasing the prediction accuracy. The ConvNet architecture that was used for classifying four classes has 13 layers which include five Conv2D layer , five Max pooling layer, one flatten layer and two fully connected (Dense) layer and It takes RGB image as input which is reshaped into (200, 200). Figure 4.9 represents the details of the 13 layer ConvNet architecture implemented in the research.

Layer	Filter Shape	Input Size	Parameter
Input		200 x 200 x 3	
Conv2D	3 x 3 x 3 x 16	198 x 198 x 16	488
Maxpooling2D	0	99 x 99 x 16	0
Conv2D	3 x 3 x 16 x 32	97 x 97 x 32	4640
Maxpooling2D	0	48 x 48 x 32	0
Conv2D	3 x 3 x 32 x 64	64 x 64 x 46	18496
Maxpooling2D	0	23 x 23 x 64	0
Conv2D	3 x 3 x 64 x 64	21 x 21 x 64	36928
Maxpooling2D	0	10 x 10 x 64	0
Conv2D	3 x 3 x 64 x 64	8 x 8 x 64	36928
Maxpooling2D	0	4 x 4 x 64	0
Flatten	1	1024	0
Dense	1028 x 128	128	131200
Dense	128 x 4	4	516

Figure 4.9: ConvNet Architecture summary for four classes

In this architecture,

Total params: 14,815,044
Trainable params: 100,356
Non-trainable params: 14,714,688

After achieving the desired results with ConvNet, some Transfer Learning models are implemented which are the extended version of CNN architecture for better results as well as for comparison. The Transfer Learning techniques which are used in this research are MobileNet, VGG16 and InceptionV3.

4.5 MobileNet implementation On four classes

The core component of MobileNet Architecture is on depth separable convolution. MobileNet factorizes standard convolution into two forms of convolution, one is depthwise convolution and another one is pointwise convolution where a 1 x 1 filter is applied. In this model, depthwize convolution implements single filter convolution on input image then the pointwise convolution combines all the output of depthwise operation. So, MobileNet's convolution layer does two operations, for both filtering and combination. This minimizes model size and complexity. The recommended input image size of MobileNet is (2244, 224, 3).Figure 4.10 represents the details of the 13 layer ConvNet architecture implemented in the research.[11]

MobileNet Architecture Summary For Four Classes

Type / Stride	Filter Shape	Input Size	Parameters
Conv / s2	3 × 3 × 3 × 32	224 × 224 × 3	384
Conv dw / s1	3 × 3 × 32 dw	112 × 112 × 32	288
Conv / s1	1 × 1 × 32 × 64	112 × 112 × 32	2048
Conv dw / s2	3 × 3 × 64 dw	112 × 112 × 64	576
Conv / s1	1 × 1 × 64 × 128	56 × 56 × 64	8192
Conv dw / s1	3 × 3 × 128 dw	56 × 56 × 128	1152
Conv / s1	1 × 1 × 128 × 128	56 × 56 × 128	16384
Conv dw / s2	3 × 3 × 128 dw	56 × 56 × 128	1152
Conv / s1	1 × 1 × 128 × 256	28 × 28 × 128	32768
Conv dw / s1	3 × 3 × 256 dw	28 × 28 × 256	2304
Conv / s1	1 × 1 × 256 × 256	28 × 28 × 256	65536
Conv dw / s2	3 × 3 × 256 dw	28 × 28 × 256	2304
Conv / s1	1 × 1 × 256 × 512	14 × 14 × 256	131072
5× Conv dw / s1 Conv / s1	3 × 3 × 512 dw 1 × 1 × 512 × 512	14 × 14 × 512 14 × 14 × 512	262144
Conv dw / s2	3 × 3 × 512 dw	14 × 14 × 512	4608
Conv / s1	1 × 1 × 512 × 1024	7 × 7 × 512	524288
Conv dw / s2	3 × 3 × 1024 dw	7 × 7 × 1024	9216
Conv / s1	1 × 1 × 1024 × 1024	7 × 7 × 1024	1048576
Avg Pool / s1	Pool 7 × 7	7 × 7 × 1024	0
Dense	1024 × 4	4	4096

Figure 4.10: MobileNet Architecture summary for four classes

In this architecture,

Total params: 3,232,964
Trainable params: 1,867,780
Non-trainable params: 1,365,184

4.6 VGG16 implementation on four Classes

VGG16 is a pre-trained transfer learning model that is mostly used to classify a large number of data sets. VGG16 also known as Very Deep Convolutional Networks. The performance of a ConvNet Model can be improved by increasing the depth of the layers. VGG16 takes a fixed input (224, 224) size and in the preprocessing stage it only subtracts the mean value of RGB. In the training stage the training set images pass through a stack of convolution layers where filters are set to 3 x 3. However, in configuration, 1 x 1 filter is used. The stride is set to 1 pixel of the image and spatial padding also set to 1 pixel for 3 x 3 convolution operation. In this model the max pooling layers operate with 2 x 2 filters and stride 2. Two fully connected layers with 25088 and 4 (for four classes) channels come after the max pooling layer.Figure 4.11 represents the details of the 13 layer ConvNet architecture implemented in the research. [12]

VGG16 Architecture Summary For Four Classes

Layer	Filter Shape	Input Size	Parameter
Input		224 x 224 x 3	
Conv1	3 x 3 x 3 x 64	224 x 224 x 64	1792
Conv2	3 x 3 x 64 x 64	224 x 224 x 64	36928
Maxpooling	0	112 x 112 x 64	0
Conv1	3 x 3 x 64 x 128	112 x 112 x 128	73856
Conv2	3 x 3 x 128 x 128	112 x 112 x 128	147584
Maxpooling	0	56 x 56 x 128	0
Conv1	3 x 3 x 128 x 256	56 x 56 x 256	295168
Conv2	3 x 3 x 256 x 256	56 x 56 x 256	590080
Conv3	3 x 3 x 256 x 256	56 x 56 x 256	590080
Maxpooling	0	28 x 28 x 256	0
Conv1	3 x 3 x 256 x 512	28 x 28 x 512	1180160
Conv2	3 x 3 x 512 x 512	28 x 28 x 512	2359808
Conv3	3 x 3 x 512 x 512	28 x 28 x 512	2359808
Maxpooling	0	14 x 14 x 512	0
Conv1	3 x 3 x 512 x 512	14 x 14 x 512	2359808
Conv2	3 x 3 x 512 x 512	14 x 14 x 512	2359808
Conv3	3 x 3 x 512 x 512	14 x 14 x 512	235808
Maxpooling	0	7 x 7 x 512	0
Flatten	1	25088	0
Dence	25088 x 4	4	100356

Figure 4.11: VGG16 Architecture summary for four classes

In this architecture ,

Total params: 14,815,044
Trainable params: 100,356
Non-trainable params: 14,714,688

4.7 InceptionV3

InceptionV3 is the 3rd version of Convolution Neural Network developed by Google. InceptionV3 is trained on ImageNet Database [42]. The recommended input image size for this model is (299, 299, 3). However, in this research, the input size per image. was (224, 224, 3). InceptionV3 has a good classification record in the BioMedical field[16].

In this architecture ,

Total params: 21,802,784
Trainable params: 11,114,880
Non-trainable params: 10,687,904

Chapter 5

Performance Analysis

The research is conducted in multiple phases in which different CNN Deep Learning Models are used. In the beginning of the research a simple Convolution model was implemented on two classes and their performance was promising. In later phases, more depth ConvNet with more layers are implemented on the four classes to achieve better accuracy. Furthermore, some extended versions of CNN architecture are used which are also known as Transfer Learning techniques which includes MobileNet, VGG16, InceptionV3. The Performance of the implemented Classification models can be measured using evaluation metrics. Confusion matrix, Accuracy, precision, recall and F1-score are among these performance measuring metrics. With the help of this measurement, the strengths and limitations of these models can be evaluated [33].

5.1 Confusion Matrix

A confusion matrix is a table that summarizes how a classification model performed on a set of test data for which the real values are known. It is one of the most often utilized performance measurement indicators in data analysis. It contains four characteristics that are employed in a variety of different computations, including F1 score, recall, precision, and accuracy. Figure 5.1 depicts a four-class confusion matrix. True Positives (TP), True Negatives (TN), False Positives (FP), and False Negatives (FN) are the four words (FN) [7].

True Positives (TP) - These are accurately predicted positive values which indicates that the value of the actual class and the value of the predicted class are both true. For example, if the actual class value shows that this patient has MI and the predicted class also suggests that this patient has MI [7].

True Negatives (TN) - These are correctly predicted negative values, implying that the value of the actual class is negative and the value of the predicted class is negative as well. For example, if the actual class states the patient does not have MI and the predicted class says the same [7].

False positives and false negatives happen when your actual class values differ from the predicted class values.

False Positives (FP) - These are situations when the actual class is not the same as the predicted class. For example, if the actual class indicates that this patient has no MI but the predicted class indicates that this patient has MI[7].

False Negatives (FN) - These are situations in which the actual class is positive but the expected class is negative. For example, if the patient's actual class value indicates that he has MI, while the predicted class value implies that the patient is Normal [7].

Proposed models for this research have used five samples from each classes for testing to build a 4x4 confusion matrix and a classification report.

Figure 5.1 represents the confusion matrix of all the models for four classes.

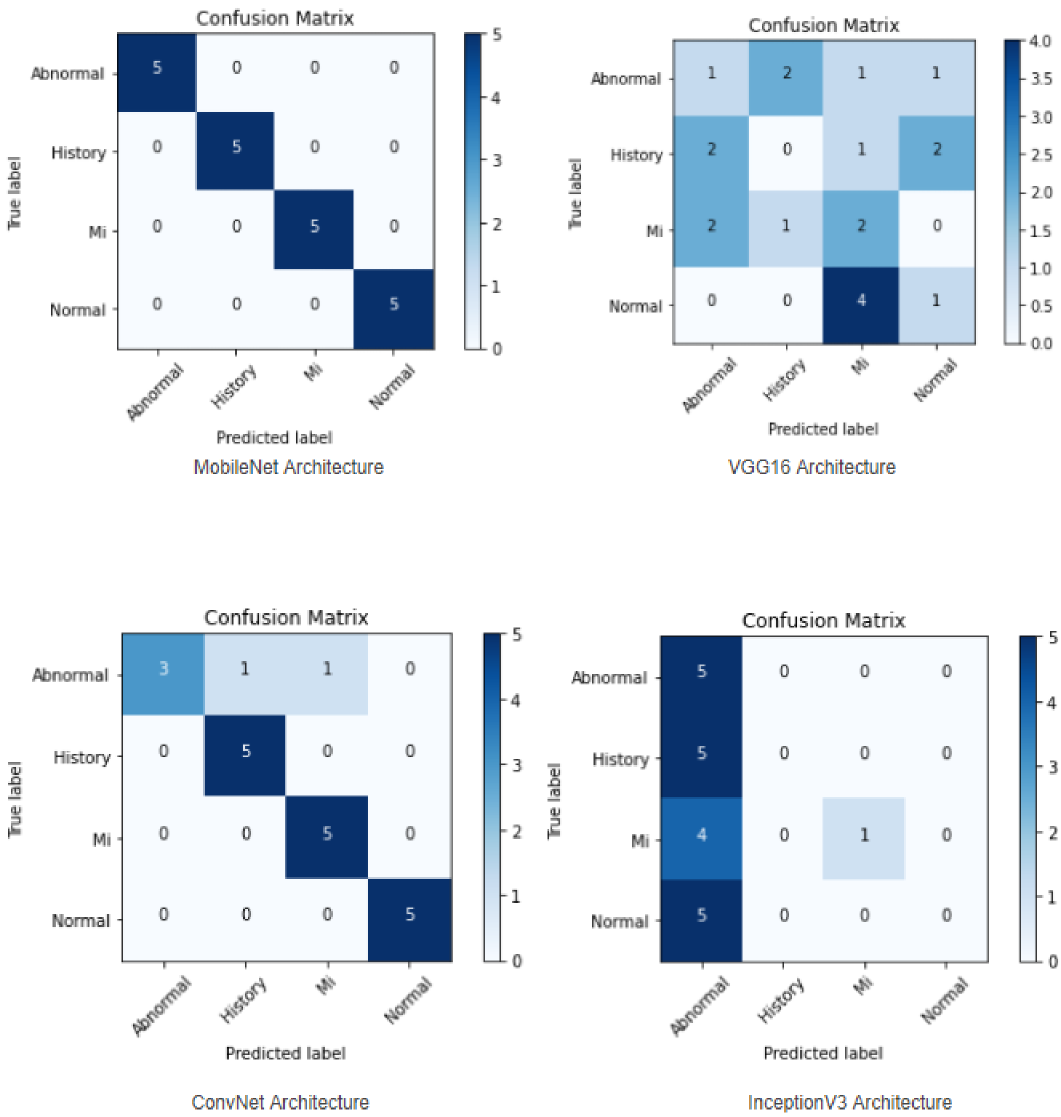

Figure 5.1: the confusion matrix of four classes

5.2 Precision Score

Precision scores reflect the ability of the model to predict the positive results out of all its positive predictions which includes true positives and false positives. When the classes are extremely imbalanced, the precision score is a good indicator of prediction success. Mathematically it represents the ratio of true positives to the sum of true positives and false positives[33].

$$PrecisionScore = \frac{TP}{(FP + TP)} \tag{5.1}$$

Figure 5.2 represents the precision score table of all the implemented models in this research.

Precision Table of All the Models used for four Classes

	MobileNet	VGG16	ConvNet	InceptionV3
Abnormal	0.83	1	1	0.26
History	0.71	1	0.83	0
Mi	1	0.62	0.83	1
Normal	1	1	1	0
Macro Average	0.89	0.91	0.92	0.32
Weighted Average	0.89	0.91	.0.92	0.32

Figure 5.2: Precision Table of All the Models used for four Classes

5.3 Recall Score

The recall score measures the model's ability to reliably forecast positives from real positives. This differs from precision as it counts how many positive predictions a model makes out of all positive predictions. In other words, it assesses the classification model based on all true positives among all possible positives in a dataset. The better the classification model is in identifying both positive and negative samples, the higher the recall score [33].Mathematically it denotes the proportion of actual positives to the sum of true positives and false negatives.

$$RecallScore = \frac{TP}{(FN + TP)} \tag{5.2}$$

Figure 5.3 represents the precision score table of all the implemented models in this research.

Recall Table of All the Models used for four Classes

	MobileNet	VGG16	ConvNet	InceptionV3
Abnormal	1	1	0.6	1
History	1	0.6	1	0
Mi	1	1	1	0.2
Normal	0.4	0.8	1	0
Macro Average	0.85	0.85	0.9	0.3
Weighted Average	0.85	0.85	0.9	0.3

Figure 5.3: Recall Table of All the Models used for four Classes

5.4 Accuracy Score

The ratio of true positives and true negatives to all positive and negative observations is defined as model accuracy. In other words, accuracy estimates the probability of the classification models accurate anticipations on outcome based on the total number of predictions [33].The accuracy rate is useful .However, it describes nothing about the inaccuracies of the classification models when dealing with new data.

In mathematics, It is the ratio of the total of true positive and true negative out of all predictions .

$$AccuracyScore = \frac{(TP + TN)}{(TP + FN + TN + FP)} \tag{5.3}$$

5.5 F1 Score

The F1 Score is the weighted average of Precision and Recall. As a result, this score takes both false positives and false negatives into account. Although not as obvious as accuracy, F1 is frequently more valuable than accuracy, particularly when class distribution is unequal [7]. The F1 score in mathematics is a weighted average of Precision and Recall.

$$F1Score = \frac{(2 * PrecisionScore * RecallScore)}{(PrecisionScore + RecallScore)} \tag{5.4}$$

Figure 5.4 represents the precision score table of all the implemented models in this research.

	MobileNet	VGG16	ConvNet	InceptionV3
Abnormal	0.91	1	0.75	0.42
History	0.83	0.75	0.91	0
Mi	1	0.77	0.91	0.33
Normal	0.57	0.89	1	0
Accuracy	0.85	0.85	0.9	0.3
Macro Average	0.83	0.85	0.89	0.19
Weighted Average	0.83	0.85	0.89	0.19

Figure 5.4: F1 Score Table of All Models used for four Classes

Chapter 6

Statistical Representation

An overfitted model is unable to generalize or make accurate predictions for data that it has not seen before since it matches the training data well. To determine whether or not the model is over-fitted, they employ a technique known as cross-validation, which divides their data into two sets: a training set and a validation set. The model is trained on the training data, whilst the validation data is used to evaluate the model's output. During the training phase, metrics on the training set are used to gauge the model's progress; metrics on the validation set are used to gauge its performance. Measures of the training and validation sets' loss and accuracy are referred to as the training and validation set's measures, respectively, in this context. The results of this study, based on this data, were positive, as shown by the models of visualizations for two different classes.

First, the accuracy on the training set and the accuracy on the validation set are represented using two different lines one for the training set and one for validation or test set. Figure 6.1 shows how the accuracy for training and validation sets increases relative to epoch.

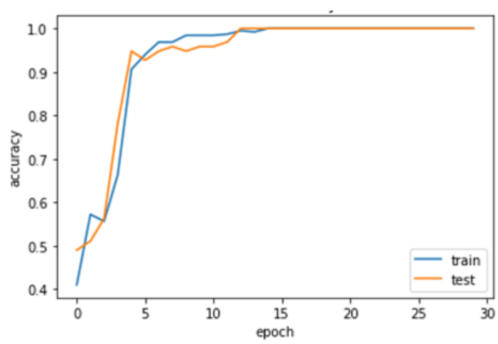

Figure 6.1: Accuracy Curve

The loss on the training set and the loss on the validation set are represented using two different lines one for train and one for validation or test set. Figure 6.2 shows how the losses for training and validation set decreased relative to epoch.

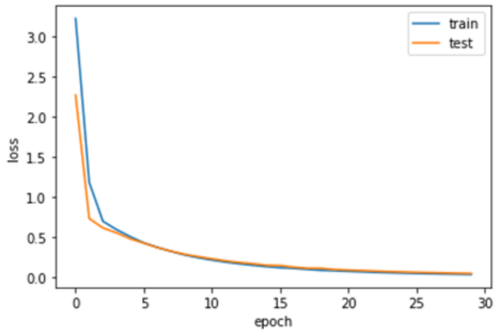

Figure 6.2: Loss Curve

In this Line model (accuracy, loss) and (val accuracy, val loss) points are plotted as dots with a line going through them for the training set as well as for the validation set .red line is for training set and green line represents the validation set,the x-axis is labeled as accuracy and y-axis is labeled as loss.From the presentation of (figure it shows at first loss was 3.0 which decreased to 0.0 for training set and for validation set the loss was less than 2.5 which decreased to 0.0 and accuracy increased to 1.0 for both of the sets. Figure 6.3 represents the line accuracy vs loss.

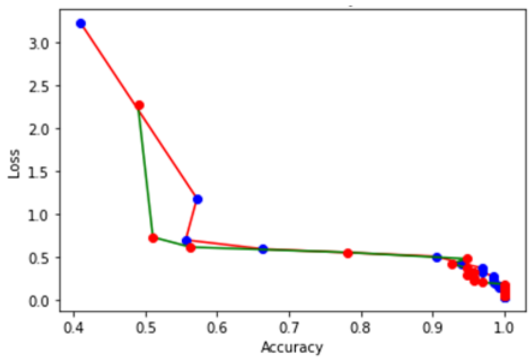

Figure 6.3: Line Accuracy vs Loss Model

6.1 13 Layer ConvNet Architecture

Furthermore,In this ConvNet 13 layer model, 548 images have been used for training and 129 images used for validation out of a total 688 images from four classes. Training accuracy determines how well the data has been trained in the model and Validation accuracy defines how well the model can classify the validation dataset. In the same way, Training Loss describes how well the ConvNet model adopts the training data and validation loss shows how well this model is fitting new data.

Figure 6.4 indicates overall accuracy of the ConvNet Architecture based on Training Accuracy, Validation Accuracy, Training Loss and Validation loss within 30 epochs.

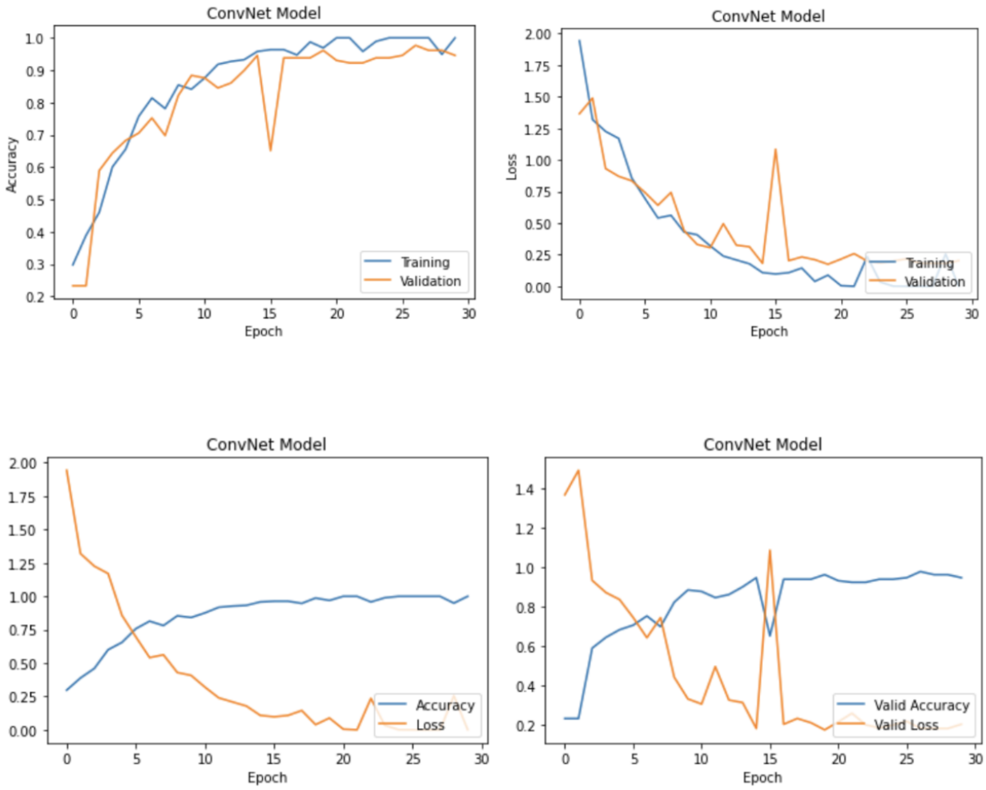

Figure 6.4: ConvNet Curves

6.2 VGG16 Architecture

VGG16 Architecture has been pre training in the ImageNet dataset. Again the model has been trained with 508 ECG images from four classes and for validation 160 images have been inserted from four classes. Like ConvNet, in VGG16 the epoch number is set to 30. The figure 6.5 describes the performance of the VGG16 Architecture.

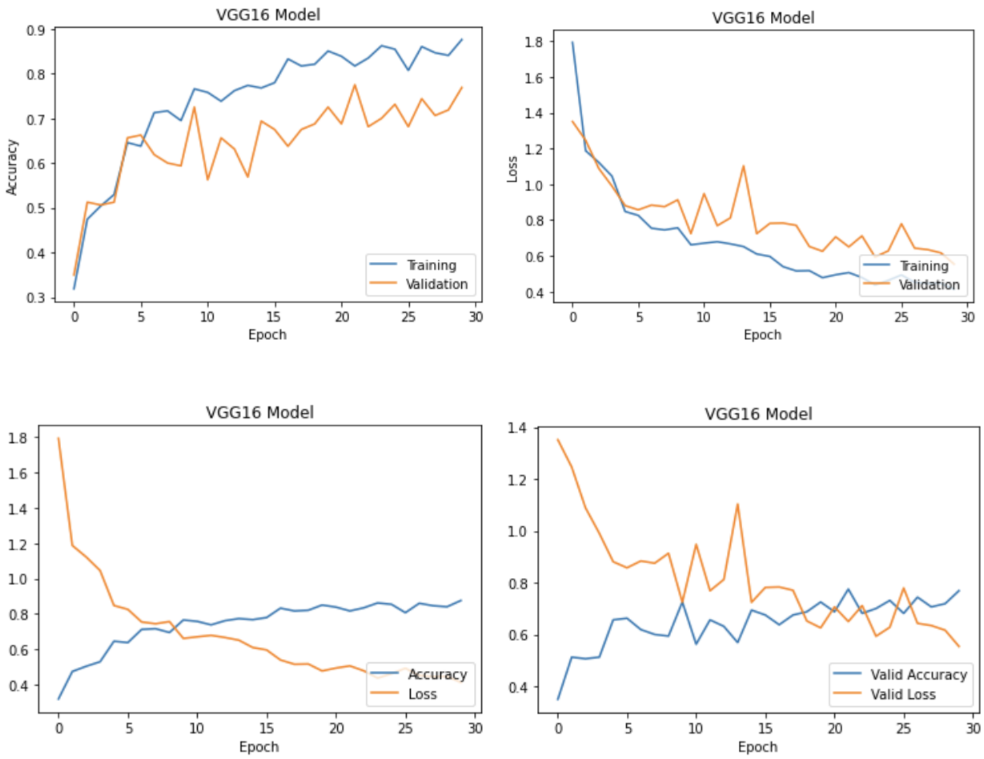

Figure 6.5: VGG16 curves

6.3 MobileNet Architecture

Just like ConvNet Architecture, in MobileNet Architecture 548 RGB images have been chosen for training and 129 RGB images for Validation. For MobileNet analysis, epoch is set to 30. In Figure 6.6 overall result of MobileNet Architecture has been described using below ROC curve.

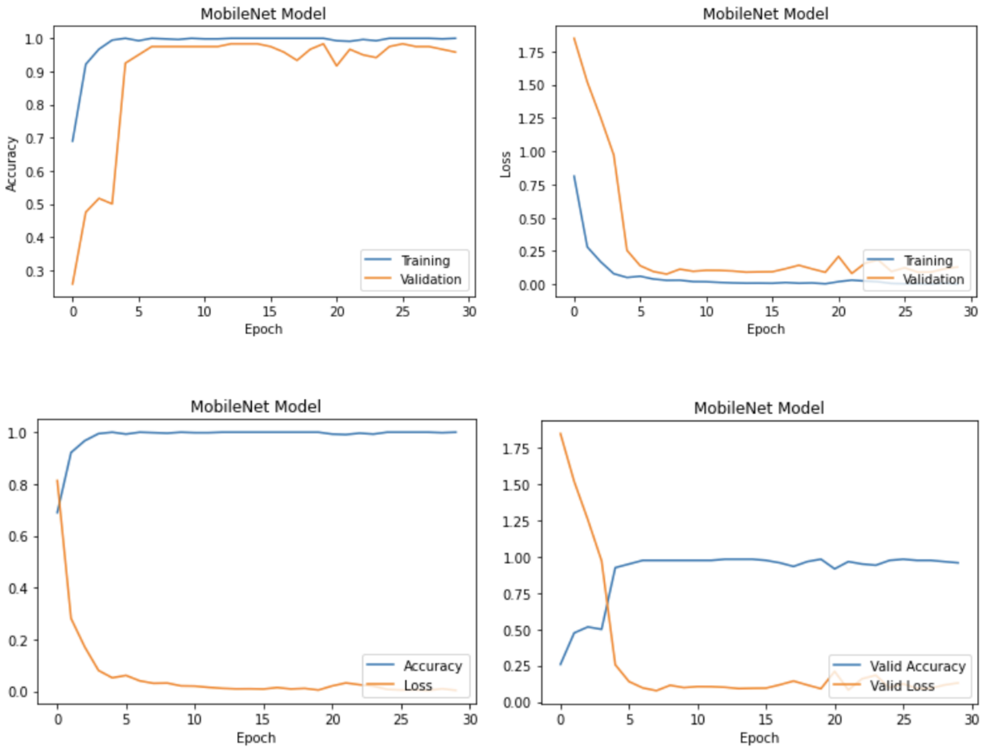

Figure 6.6: MobileNet curves

6.4 InceptionV3 Architecture

The dataset has been splitted into 548 for training and 129 for validation, just like previously mentioned models. However, 40 epochs have been applied for better accuracy. Figure 6.7, analyze the model on Training Accuracy, Validation Accuracy, Training Loss and Validation loss within 30 epochs.

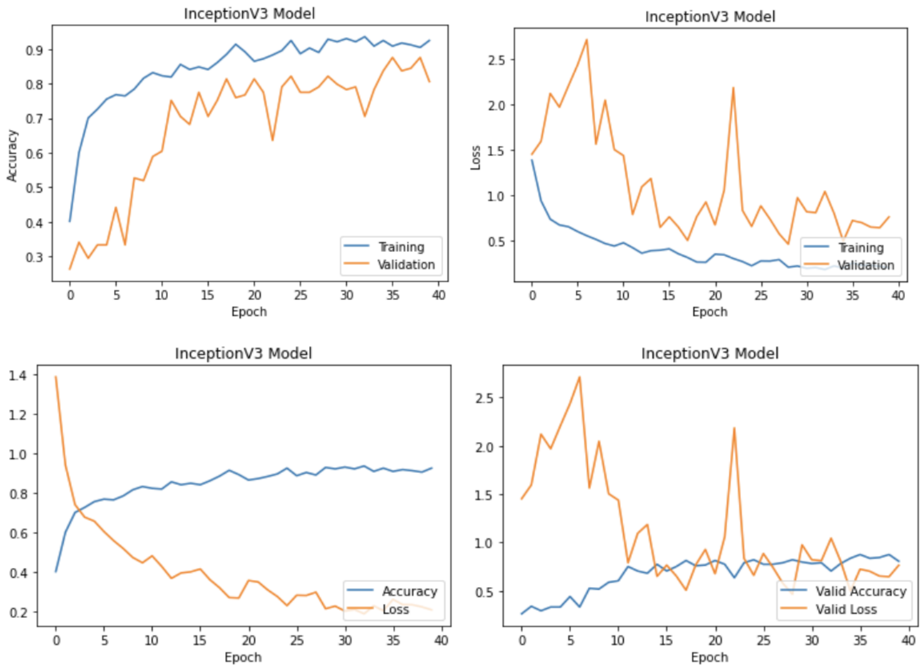

Figure 6.7: InceptionV3 curves

With all this comparing visualization representation, the result of the analysis is quite clear that the model works in the most optimal way for both of the data sets of cross verification they both give accuracy of 1.00 around 15 epoch which is promising also it means the model can classify any unknown data with approximately 100 percent accuracy which is a tremendous achievement.

Chapter 7

Result

After running the models using Google Colab and Anaconda Jupyter Notebook, results of prediction and detection of data are obtained. Some ECG images were collected from a dataset in .jpg file format from a Pakistani hospital who aims to help the scientific community with this information for the advancement of Al technologies and also to conduct research on Cardiovascular diseases. The dataset that we used for this research consists of 688 ECG images which were later on resized, normalized and labeled accordingly as well as the data were prepossessed suitable according to their requirements . Then those data were splitted for train, validation and test.These data were fitted to multiple classification models which includes ConvNet, MobileNet, VGG16, InceptionV3 and gets predictions. For getting the prediction right Probability is used and based on the max probability of all four classes the sample data is predicted.Figure 7.1 represents an abnormal ECG signal.

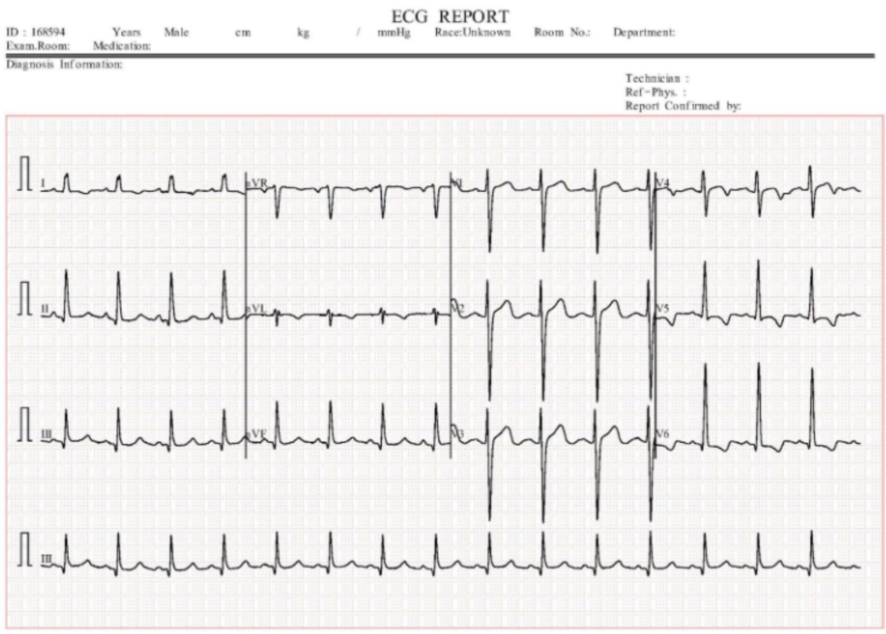

Figure 7.1: Abnormal ECG Signal

this above figure:7.1, the probabilities of the image of the four classes based on the Abnormal analysis of each proposed Model is given below in table 7.1,

Table 7.1: Abnormal ECG signal

	Abnormal	History	Mi	Normal
MobileNet	99.95	0.040	0.002	0.004
VGG16	69.63	27.355	0.0907	2.915
ConvNet	99.99	5.9802879e^{-}06	2.0604986e^{-}05	5.1263531e^{-}07
InceptionV3	99.99	7.1915201e^{-}04	6.7390571e^{-}8	5.3756355e^{-}07

Similarly,Figure 7.2,Represents a history of Patients ECG signal report

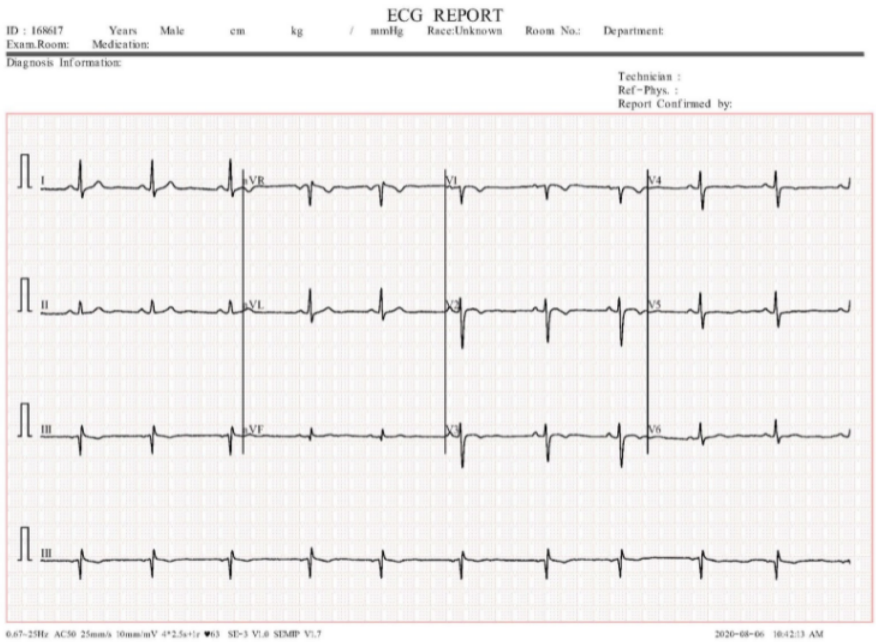

Figure 7.2: represents an History ECG signal

the probabilities of the image of the four classes based on the History analysis of each proposed Model is given below in table

Table 7.2: MI History ECG signal

	Abnormal	History	Mi	Normal
MobileNet	2.391e-4	99.97	6.275e^{-}05	0.002
VGG16	96.70	1.429	0.918	0.940
ConvNet	2.4593785e($^{-}$8)	100.00	6.8600632e^{-}20	2.2758782e^{-}07
InceptionV3	70.908	29.083	0.001	0.007

Figure 7.3: represents MI ECG signal

For this above figure:7.3, the probabilities of the image of the four classes based on the MI analysis of each proposed Model is given below in table 7.3,

Table 7.3: MI ECG signal

	Abnormal	History	Mi	Normal
MobileNet	8.4970779e-04	0.008	99.98	2.9006096e-04
VGG16	96.70	14.29	0.91	0.944
ConvNet	4.2703054e-16	5.0351261e-23	100.00	2.0146889e-12
InceptionV3	96.69	3.303	2.6939195e-8	1.0255055e-06

Figure 7.4: Normal ECG signal

For this above Figure:7.4, the probabilities of the image of the four classes based on the normal analysis of each proposed Model is given below in table 7.4,

Table 7.4: Normal ECG signal

	Abnormal	History	Mi	Normal
MobileNet	0.001	0.028	0.001	99.96
VGG16	65.87	22.52	0.0325	11.274
ConvNet	4.5431739e-05	6.5277213e-06	1.3647897e-13	99.99
InceptionV3	99.74	0.257	1.0396383e-07	2.0468133e-05

Table 7.5 represents the comparison of CNN models

Table 7.5: Comparison of CNN models

Model	**Precision**	**Recall**	**f1 score**	**Accuracy**
ConvNet	0.92	0.9	0.89	0.9
VGG16	0.91	0.85	0.85	0.85
MobileNet	0.89	0.85	0.83	0.85
InceptionV3	0.32	0.3	0.19	0.3

Comparing the above analysis, all models have been trained successfully and some of the models have shown significant results. MobileNet and ConvNet architecture classified the Abnormal, Mi, History and Normal with high prediction rate. Based on the max probability the data is categorized.

Chapter 8

Conclusion

This paper has addressed a novel approach to detecting Myocardial Infarction by using 12-lead ECG signals. Using multiple deep learning approaches, this study presents a unique strategy for identifying Myocardial Infarction (MI). It is one of the most commonly seen cardiovascular diseases worldwide, with a high fatality rate if it is not detected and diagnosed on time. The purpose of the study is to build a model that can contribute to the advancement of seeing MI in an instant with the help of advanced Artificial Intelligence. The research contains the ConvNet model and other prominent transfer learning models like MobileNet, VGG16, and InceptionV3, which take 12-lead ECG data as input. Compared to VGG16 and InceptionV3, the trained model using the proposed ConvNet and MobileNet architecture showed good accuracy in MI diagnosis. The suggested models' performance is evaluated using the confusion matrix, Precision score, F1-score, Recall score, and ROC curve matric. The accuracy is average 97.50 percent, obtained by utilizing MobileNet and the ConvNet model. The VGG16 and InceptionV3 showed good accuracy while training the samples. However, in the validation and test phase, the results were not compatible. In conclusion, the proposed model has a promising MI detection performance which can be used in intensive care units as well as be accessible to general people.

References

[1] M. J. Wolk, S. Scheidt, and T. Killip, "Heart failure complicating acute myocardial infarction," en, *Circulation*, vol. 45, no. 5, pp. 1125–1138, 1972.

[2] A. Rashid, Zahooruddin, I. M. Qureshi, and A. Saleem, "Electrocardiogram signal processing for baseline noise removal using blind source separation techniques: A comparative analysis," in *2011 International Conference on Machine Learning and Cybernetics*, IEEE, 2011.

[3] P. Valensi, L. Lorgis, and Y. Cottin, "Prevalence, incidence, predictive factors and prognosis of silent myocardial infarction: A review of the literature," en, *Arch. Cardiovasc. Dis.*, vol. 104, no. 3, pp. 178–188, 2011.

[4] J. G. Jollis, C. B. Granger, T. D. Henry, *et al.*, "Systems of care for ST-segment-elevation myocardial infarction: A report from the american heart association's mission: Lifeline," en, *Circ. Cardiovasc. Qual. Outcomes*, vol. 5, no. 4, pp. 423–428, 2012.

[5] P. T. O'Gara, F. G. Kushner, D. D. Ascheim, *et al.*, "2013 ACCF/AHA guideline for the management of ST-elevation myocardial infarction: Executive summary: A report of the american college of cardiology Foundation/American heart association task force on practice guidelines: A report of the american college of cardiology foundation/american heart association task force on practice guidelines," en, *Circulation*, vol. 127, no. 4, pp. 529–555, 2013.

[6] V. Kumar, A. Abbus, and J. Aster, "Robbins and cotran pathologic basis of disease," *Elsevier health sciences*, 2014.

[7] *Accuracy, precision, recall & F1 score: Interpretation of performance measures*, en, https://blog.exsilio.com/all/accuracy-precision-recall-f1-score-interpretation-of-performance-measures/?fbclid=IwAR2fNgJVXl_kchYz1YLGVKtnT_yc0BVGVT2ZwUpfYI_2mb7Gkf9QnB1FBDg, Accessed: 2022-1-16, Sep. 2016.

[8] T. Vos, C. Allen, M. Arora, *et al.*, "Global, regional, and national incidence, prevalence, and years lived with disability for 310 diseases and injuries, 1990–2015: A systematic analysis for the global burden of disease study 2015," *Lancet*, vol. 388, no. 10053, pp. 1545–1602, 2016.

[9] M. Abdelazez, P. X. Quesnel, A. D. C. Chan, and H. Yang, "Signal quality analysis of ambulatory electrocardiograms to gate false myocardial ischemia alarms," *IEEE Trans. Biomed. Eng.*, vol. 64, no. 6, pp. 1318–1325, 2017.

[10] S. Albawi, T. A. Mohammed, and S. Al-Zawi, "Understanding of a convolutional neural network," in *2017 International Conference on Engineering and Technology (ICET)*, IEEE, 2017.

[11] A. G. Howard, M. Zhu, B. Chen, *et al.*, "MobileNets: Efficient convolutional neural networks for mobile vision applications," 2017. eprint: 1704.04861.

[12] ——, "MobileNets: Efficient convolutional neural networks for mobile vision applications," 2017. eprint: 1704.04861.

[13] B. Pyakillya, N. Kazachenko, and N. Mikhailovsky, "Deep learning for ECG classification," *J. Phys. Conf. Ser.*, vol. 913, p. 012 004, 2017.

[14] A. K. Dohare, V. Kumar, and R. Kumar, "Detection of myocardial infarction in 12 lead ECG using support vector machine," en, *Appl. Soft Comput.*, vol. 64, pp. 138–147, 2018.

[15] S. Khan, H. Rahmani, S. A. A. Shah, and M. Bennamoun, "A guide to convolutional neural networks for computer vision," *Synth. lect. comput. vis.*, vol. 8, no. 1, pp. 1–207, 2018.

[16] L. D. Nguyen, D. Lin, Z. Lin, and J. Cao, "Deep CNNs for microscopic image classification by exploiting transfer learning and feature concatenation," in *2018 IEEE International Symposium on Circuits and Systems (ISCAS)*, IEEE, 2018.

[17] Ö. Yıldırım, P. Pławiak, R.-S. Tan, and U. R. Acharya, "Arrhythmia detection using deep convolutional neural network with long duration ECG signals," en, *Comput. Biol. Med.*, vol. 102, pp. 411–420, 2018.

[18] A. B. Abdul Qayyum, T. Islam, and M. A. Haque, "ECG heartbeat classification: A comparative performance analysis between one and two dimensional convolutional neural network," in *2019 IEEE International Conference on Biomedical Engineering, Computer and Information Technology for Health (BECITHCON)*, IEEE, 2019.

[19] U. R. Acharya, H. Fujita, S. L. Oh, *et al.*, "Deep convolutional neural network for the automated diagnosis of congestive heart failure using ECG signals," en, *Appl. Intell.*, vol. 49, no. 1, pp. 16–27, 2019.

[20] U. B. Baloglu, M. Talo, O. Yildirim, R. S. Tan, and U. R. Acharya, "Classification of myocardial infarction with multi-lead ECG signals and deep CNN," en, *Pattern Recognit. Lett.*, vol. 122, pp. 23–30, 2019.

[21] J. Jeong, *The most intuitive and easiest guide for convolutional neural network*, en, https://towardsdatascience.com/the-most-intuitive-and-easiestguide-for-convolutional-neural-network-3607be47480, Accessed: 2022-1-12, Jan. 2019.

[22] R. Kher and G. H. Patel, *Signal processing techniques for removing noise from ECG signals*, http://www.jscholaronline.org/articles/JBER/Signal-Processing.pdf, Accessed: 2022-1-12, 2019.

[23] S. Saxena, R. Jais, and M. K. Hota, "Removal of powerline interference from ECG signal using FIR, IIR, DWT and NLMS adaptive filter," in *2019 International Conference on Communication and Signal Processing (ICCSP)*, IEEE, 2019.

[24] S. W. Smith, B. Walsh, K. Grauer, *et al.*, "A deep neural network learning algorithm outperforms a conventional algorithm for emergency department electrocardiogram interpretation," en, *J. Electrocardiol.*, vol. 52, pp. 88–95, 2019.

[25] R. Avanzato and F. Beritelli, "Automatic ECG diagnosis using convolutional neural network," en, *Electronics (Basel)*, vol. 9, no. 6, p. 951, 2020.

[26] ——, "Automatic ECG diagnosis using convolutional neural network," en, *Electronics (Basel)*, vol. 9, no. 6, p. 951, 2020.

[27] O. S. Lih, V. Jahmunah, T. R. San, *et al.*, "Comprehensive electrocardiographic diagnosis based on deep learning," en, *Artif. Intell. Med.*, vol. 103, no. 101789, p. 101 789, 2020.

[28] P. Natesan, V. V. Priya, and E. Gothai, "Classification of Multi-Lead ECG signals to predict myocardial infarction using CNN," in *IEEE. 2020 Fourth International Conference on Computing Methodologies and Communication (ICCMC)*, 2020, pp. 1029–1033.

[29] E. Prabhakararao and S. Dandapat, "Myocardial infarction severity stages classification from ECG signals using attentional recurrent neural network," *IEEE Sens. J.*, vol. 20, no. 15, pp. 8711–8720, 2020.

[30] CDC, *Heart disease facts*, en, https://www.cdc.gov/heartdisease/facts.htm, Accessed: 2022-1-12, Sep. 2021.

[31] A. H. Khan, *ECG images dataset of cardiac patients*, Mar. 2021.

[32] A. H. Khan, M. Hussain, and M. K. Malik, "Cardiac disorder classification by electrocardiogram sensing using deep neural network," en, *Complexity*, vol. 2021, pp. 1–8, 2021.

[33] A. Kumar, *Accuracy, precision, recall & f1-score - python examples*, en, https://vitalflux.com/accuracy-precision-recall-f1-score-python-example/?fbclid=IwAR3MGfZE4PN_lbT0MexklF55HqYO7eTYV1BTT6SrNADEWUAFd36EnV12UpE, Accessed: 2022-1-16, Oct. 2021.

[34] Y. Sattar and M. Chadi Alraies, *Ventricular Aneurysm.* StatPearls Publishing, 2021.

[35] https://www.heart.org/en/health-topics/heart-attack/about-heart-attacks/heartattack-or-sudden-cardiac-arrest-how-are-they-different, Accessed: 2022-1-12.

[36] http://introtodeeplearning.com/2020/slides/6S191_MIT_DeepLearning_L3.pdf, Accessed: 2022-1-12.

[37] *Cardiogenic shock*, en, https://www.nhlbi.nih.gov/health-topics/cardiogenic-shock, Accessed: 2022-1-12.

[38] *Cardiovascular diseases (CVDs)*, en, https://www.who.int/news-room/fact-sheets/detail/cardiovascular-diseases-(cvds), Accessed: 2022-1-12.

[39] *CS231n convolutional neural networks for visual recognition*, https://cs231n.github.io/, Accessed: 2022-1-12.

[40] *Heart attack*, en, https://www.nhlbi.nih.gov/health-topics/heart-attack, Accessed: 2022-1-12.

[41] *Heart disease*, en, https://www.mayoclinic.org/diseases-conditions/heartdisease/symptoms-causes/syc-20353118, Accessed: 2022-1-12.

[42] *Inception V3 deep convolutional architecture for classifying acute*, en, https://www.intel.com/content/www/us/en/developer/articles/technical/inception-v3-deep-convolutional-architecture-for-classifying-acute-myeloidlymphoblastic.html, Accessed: 2022-1-16.

[43] J. Kolář, F. C. Ghesu, J. Lu, *et al.*, *DeepAI: The front page of A.I*, https://deepai.org/machine-learning-glossaryand-terms/stride, Accessed: 2022-1-12.

[44] M. Vijayavanan, V. Rathikarani, and P. Dhanalakshmi, *Automatic classification of ECG signal for heart disease diagnosis using morphological features*, http://www.ijcset.com/docs/IJCSET14-05-04-168.pdf, Accessed: 2022-1-12.

[45] *Why must a nonlinear activation function be used in a backpropagation neural network?* https://stackoverflow.com/questions/9782071/why-must-a-nonlinearactivation-function-be-used-in-a-backpropagation-neural-net, Accessed: 2022-1-12.