Heart Arrhythmias Detection and Classification Using ECG Signals and Deep Convolutional Neural Networks

By

Wafa Mohammed Omer Abdoallah

In Partial Fulfilment of the Requirements for the Degree of Master of Science in Biomedical Engineering (Bioinstrumentation and Imaging)

Center of Biomedical Engineering

Addis Ababa Institute of Technology

Addis Ababa University

Advisor: Dawit Assefa Haile (PhD)

Co Advisor: Melkamu Hunegnaw (MSc)

Addis Ababa, Ethiopia

December, 2021

Declaration

By submitting this MSc thesis, I declare that the entirety of the work presented in this document is my original work. Where information has been derived from other sources, I confirm that this has been indicated in the thesis and that I have not previously in its entirety or in part submitted it for obtaining any qualification.

Name: Wafa Mohammed Omer Abdoallah

Signature: _____

Date: _____

This MSc thesis has been submitted for examination with my approval as an advisor.

_____

Dawit Assefa Haile (PhD)

Certificate of Examination

Addis Ababa University

School of Graduate Studies

This is to certify that the thesis prepared by Wafa Mohammed Omer Abdoallah entitled: "Heart Arrhythmias Detection and Classification Using ECG signals and Deep Convolutional Neural Networks" submitted in partial fulfilment of the requirements for the Degree of Master of Science in Biomedical Engineering (Bioinstrumentation and Imaging) complies with the regulations of the University and meets the accepted standards with respect to originality and quality.

Signed by the examining committee

Examiner ..... Signature ..... Date .....

Examiner ..... Signature ..... Date .....

Advisor ..... Signature ..... Date .....

Co-advisor ..... Signature ..... Date .....

.....

Chief of Department or Graduate Program Coordinator

Abstract

Heart Arrhythmias Detection and Classification Using ECG signals and Deep Convolutional Neural Networks

According to the World Health Organization, almost 17 million people die each year as a result of cardiovascular illness. The irregularity and abnormalities of heartbeat rhythm which is known as arrhythmia is one of the conditions that can affect the cardiovascular system. Electrocardiogram (ECG) is a reliable tool that can be used for monitoring the cardiovascular health. Recently, classifying the ECG signals based on Artificial Intelligence (AI) is increasingly being studied. Convolutional Neural Networks (CNN) in particular have been effectively applied for the classification of ECG signals. Although high prediction accuracies have been reported, majority of previous studies have only been developed to classify limited number of arrhythmias. The methods were developed to evaluate all major types of arrhythmias using 1-D CNN to classify time domain representation of ECG waveforms. However, using 1-D CNNs has limited flexibility due to the use of 1-D kernels. There are methods reported to transform the time series signals into 2-D images using STFT and use 2-D CNN. However, STFT is difficult to apply to non stationary signals; there is no way to resolve the complete frequency content of such signals with a single localizing window size. To overcome this obstacle of Fourier decomposition, the Continuous Wavelet Transform (CWT) could be used to breakdown a signal into wavelets with a high degree of temporal localization. The S-transform could be another option since it takes the advantage of STFT and wavelet. This thesis study uses CNN classifiers for detecting and classifying heart arrhythmias based on analysis of ECG signals in time-frequency domain. The used data were extracted from a subset of MIT-BIH arrhythmia data set, that contain 1000 ECG signals of 17 classes in total, collected from 45 patients. 12 classes were chosen from the subset which include Normal Sinus Rhythm (NSR), Atrial Premature Beat (APB), Atrial Flutter (AFL), Atrial Fibrillation (AFIB), Supraventricular Tachyarrhythmia (SVTA), Premature Ventricular Contraction (PVC), Ventricular Tachycardia (VT), Idioventricular Rhythm (IVR), Ventricular Flutter (VFL), Left Bundle Branch Block Beat (LBBB), Right Bundle Branch Block Beat (RBBB), and Pacemaker Rhythm (PR). The one dimensional ECG signals were transformed into joint time frequency spectrograms using Stockwell transform and into scalograms using Continuous Wavelet Transform (CWT). By using different pretrained networks for classifying spectrograms and scalograms namely GoogleNet, SqueezeNet, and ResNet-50, different results were achieved. GoogleNet pretrained network showed the best

performance when using CWT generated scalograms with 93.85% accuracy, 96.42% precision, 84.14% sensitivity, 99.36% average specificity and 89.86 F1-score. Based on the results, transfer learning especially GoogleNet proved to be efficient in classifying the two-dimensional scalograms of cardiac arrhythmias, while reducing the burden of training network from scratch makes it easily applicable. Compared with recent techniques, results obtained using the proposed technique show the great promises of the 2-D CNN model in accurate classification of arrhythmias using CWT and S-transform and the proposed method resulted in higher accuracy and F1-score.

Keywords: Cardiac Arrhythmias, Electrocardiogram, Deep Learning, Convolutional Neural Network, Stockwell Transform, Continuous Wavelet Transform.

Acknowledgements

First and foremost, I would like to thank God for the strength and his blessing in completing this thesis.

Next, I would like to express my sincere gratitude to African Biomedical Engineering Mobility (ABEM) scholarship program, funded by the Intra-Africa Academic Mobility Scheme of the European Union for sponsoring my studies. Further, I would like to thank my supervisor Dr. Dawit Assefa Haile and co-supervisor Mr. Melkamu Hunegnaw for the continuous advice, and mentorship throughout the process of researching and writing this thesis.

I also would like to say thanks to all my friends for all the fun time and support in these two years, especially ABEM students: Degel Sharon, Pius Nuwabiine, and Jakuma Stephen.

Finally, it is my great deep appreciation to my parents for their unlimited support and continuous encouragement throughout the journey. This accomplishment would not have been possible without them. Sincere thanks to all.

Table of Contents

List of Figures

List of Tables

List of Abbreviations

1D	One Dimensional
2D	Two Dimensional
3D	Three Dimensional
AC	Accuracy
AFIB	Atrial Fibrillation
AFL	Atrial Flutter
AI	Artificial Intelligence
ANN	Artificial Neural Network
APB	Atrial Premature Beat
CA	Cardiac Arrhythmias
CNN	Convolutional Neural Network
ConvLayer	Convolutional Layer
CVD	Cardiovascular Disease
CWT	Continuous Wavelet Transform
DL	Deep Learning
ECG	Electrocardiogram
FC	Fully Connected layers
FN	False Negative
FP	False Positive
FT	Fourier Transform
IVR	Idio-ventricular Rhythm
LBBB	Left Bundle Branch Block Beat
MATLAB	Matrix Laboratory
MIT-BIH	Massachusetts Institute of Technology Beth Israel Hospital
ML	Machine Learning
MLII	Modified Limb Lead II
NSR	Normal Sinus Rhythm
PR	Pacemaker Rhythm
PVC	Premature Ventricular Contraction
RBBB	Right Bundle Branch Block Beat

ReLU	Rectified Linear Units layers
SEN	Sensitivity
SPE	Specificity
STFT	Short Time Fourier Transform
S-transform	Stockwell Transform
SVTA	Supraventricular Tachyarrhythmia
TN	True Negative
TP	True Positive
V1-V5	Precordial Leads
VFL	Ventricular Flutter
VT	Ventricular Tachycardia
WHO	World Health Organization

CHAPTER 1

Introduction

1.1 Background

The term cardiovascular refers to the system that supplies the body with blood incorporating the heart, arteries, veins, and capillaries. The different conditions that affect the heart are collectively termed cardiovascular diseases (CVDs) which include arrhythmia, coronary artery disease, heart attack, heart failure, and others [**1**]. Cardiac arrhythmia is considered as an important social issue and one of the most widespread sources of CVDs. It is a disease of the heart conduction system where there is deviation of the heartbeats from their regular beating pattern. It happens when the natural pacemaker of the heart develops an abnormal rhythm, the normal conduction pathway is interrupted, or another part of the heart works as pacemaker [**2**].

According to the World Health Organization (WHO), cardiovascular diseases are the leading cause of human mortality. Annually, over 17 million people lose their lives worldwide due to cardiovascular diseases [**3**]. Total CVD cases nearly doubled from 271 million in 1990 to 523 million in 2019, and CVD mortality rates increased from 12.1 million in 1990 to 18.6 million in 2019 [**4**]. Figure 1 and Figure 2 depict the mortality rate from cardiovascular diseases worldwide and in Africa, respectively.

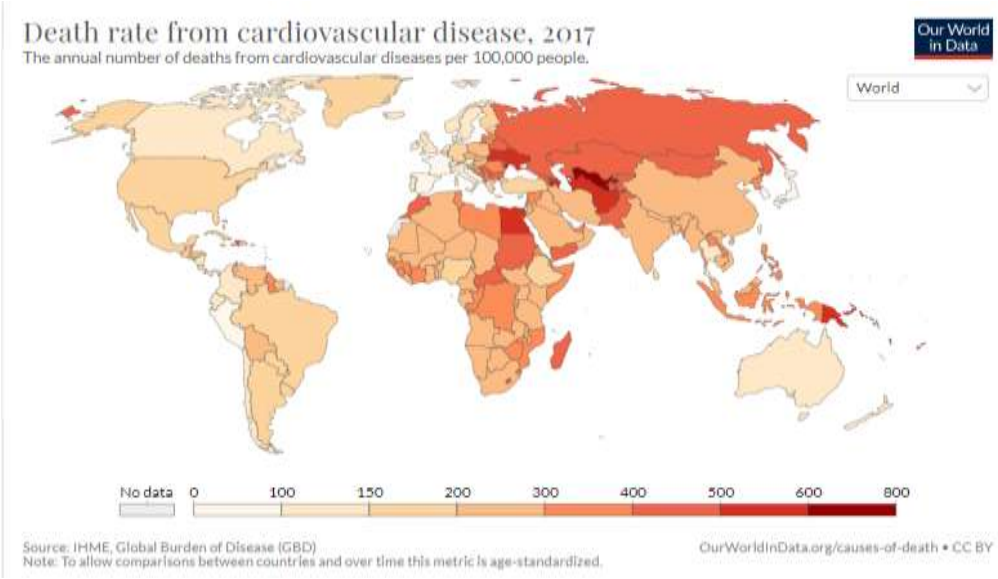


Figure 1 Death rate from cardiovascular disease 2017

Adapted from: https://ourworldindata.org/grapher/cardiovascular-disease-death-rates

(**Accessed on; Jun 23rd 2021**)

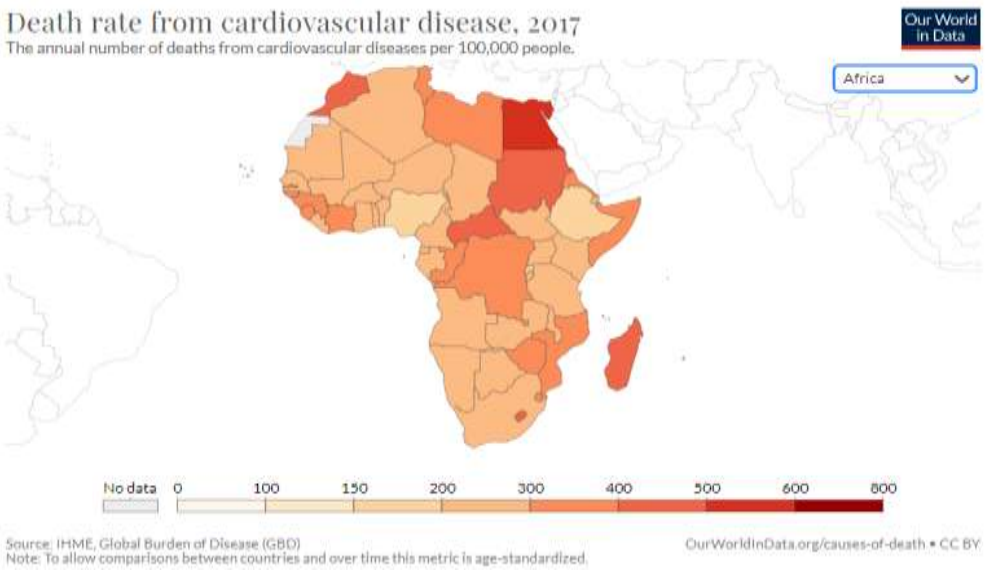


Figure 2 Death rate from cardiovascular disease, Africa 2017

Adapted from: https://ourworldindata.org/grapher/cardiovascular-disease-death-rates?region=Africa

(**Accessed on; Jun 23rd 2021**)

Arrhythmia affects millions of people [**5**]. The death resulted from the two types of arrhythmia, atrial fibrillation and atrial flutter, increased from 29,000 in 1990 up to 112,000 in 2013 [**6**]. About 80% of sudden cardiac deaths result from ventricular arrhythmias [**7**]. Rapid and accurate assessment of cardiovascular diseases is important to allow physicians determine the appropriate treatment for their patients. By providing timely treatment and utilizing methods that could identify CVDs at their early stage, the mortality can be effectively reduced [**8**]. Typically, Electrocardiogram (ECG) is the most used technique for diagnosing heart's arrhythmias.

The importance of an automatic algorithm for arrhythmias classification is being increasingly recognized. Machine learning (ML) is a subgroup of artificial intelligence (AI) used for the prediction and diagnosis of several disease types [**9**]. Intuitively speaking, AI is defined as the study of any agent or device that can perceive and understand its surroundings and take appropriate action accordingly to maximize the chances of objectives achieving [**10**]. Deep learning is a subgroup of Machine Learning, used in the prediction of mortal diseases such as CVDs. Artificial neural network (ANN) is one known framework for executing AI algorithms [**11**]. It is an interconnected network of neurons and has weighted communication channels between neurons [**12**]. It can generate outputs as responses to stimuli. Figure 3 depicts the interlink between artificial intelligence, machine learning and deep learning. For classifying ECG signals, statistical approaches [**13**], and neural network approaches (ANN) [**14**] are the commonly used methods. Compared to statistical approaches, ANN uses

activation function which introduces non-linearity to the network via the hidden layer, this can allow the network to process more complex patterns [**15**].

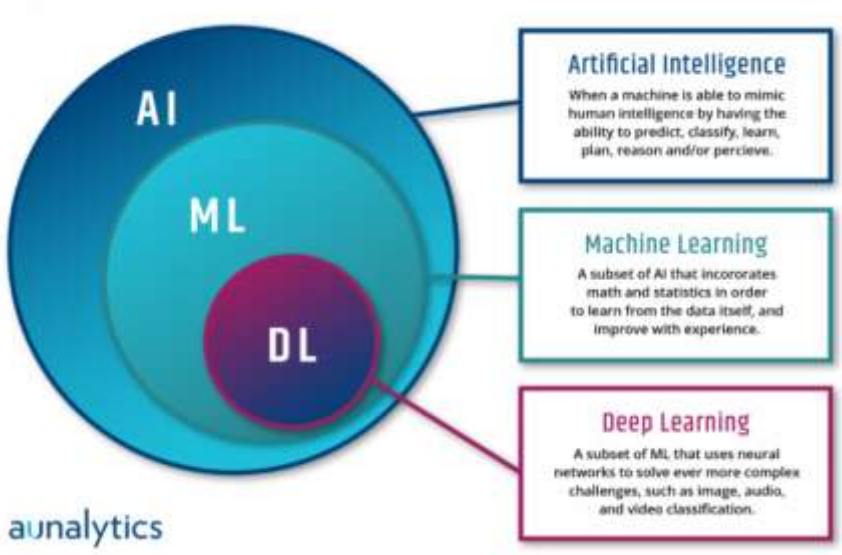


Figure 3 The Link between AI, ML and DL

Adapted from:https://www.aunalytics.com/artificial-intelligence-machine-learning-and-deep-learning%E2%81%A0/

(**Accessed on; Nov 15th 2020**)

Convolutional neural network (CNN) is a type of neural network that is used to analyze inputs using a specified grid-like architecture. CNNs can take an input image, give relevance (learnable weights and biases) to various aspects/objects in the image, and distinguish between them. In comparison to other classification algorithms, CNNs require substantially less preprocessing as it doesn't require feature extraction [**16**]. The goal of CNN is to compress the images into a format that is easier to process while preserving elements that are important for obtaining a decent prediction. The following layers make up the deep CNN: Convolution, Rectified Linear Units (ReLU), Pooling, Fully Connected, and Softmax [**17**].

Different methodologies were investigated in the neural networks sector in order to get towards classification processes. Transfer learning distinguishes out among the different current neural network algorithms in terms of time savings and versatility. Transfer learning allows a model built on related data to be utilized for another feature space [**18**]. Training a deep CNN from scratch is time consuming and needs large amount of data. When there is no enough data available for training, it's preferable to use existing neural networks that have been trained on huge data sets.

1.2 Statement of the Problem

The increasing number of people with heart diseases coupled with the time consuming procedure in classifying different heart arrhythmias is the major challenge which can lead to increased death rate. Cardiologists, before recommending any specific medication, they need to identify the type of irregular heartbeat accurately. This might be time consuming process, and cardiologists sometimes misdiagnose the type of cardiac arrhythmias [**19**]. To overcome these challenges, computer-aided diagnostic systems are needed to identify the signals automatically [**20**].

It is important to have a classifier to identify the cardiovascular diseases at their early stage [**8**]. Arrhythmia is one of the widespread sources of CVDs; where the heart is following irregular rhythm. That could be bradycardia, tachycardia, skipping a beat, cardiac arrest, atrial fibrillation, atrial flutter, and ventricular fibrillation. Classifying these different heart arrhythmias accurately could assist in more effective heart diseases diagnosis and treatment [**21**]. In that regard, there have been various efforts previously reported to automate the process of classifying the different arrhythmias. Although high prediction accuracies (more than 99%) have been reported in previous studies [**21**] [**22**], majority of them have only been tested or developed to classify limited number of arrhythmias. There are some methods used to evaluate major types of arrhythmias [**23**]. These studies used 1-D CNN to classify time domain representation of ECG waveforms. However, using 1-D CNNs has limited flexibility compared to 2-D CNNs due to the use of 1-D kernels. Hence, using time-frequency representation of ECG signals could help in better arrhythmias classification.

Short-time Fourier transform (STFT) could be used to transform the time series signals into 2-D images, as it has been used for the study which has been done for classifying eight types of arrhythmias [**21**]. However, there is no way to resolve the complete frequency content of non-stationary signals with a single window size that is used in STFT. To overcome this obstacle of Fourier decomposition, Continuous Wavelet Transform (CWT) could be used to breakdown a signal into tiny oscillations (wavelets) with a high degree of temporal localization. Furthermore, S-transform could be another option since it takes the advantage of STFT and wavelet [**24**] [**25**]; by providing a good frequency resolution at low frequency and a good time resolution at a higher frequency, because of use of frequency-dependent window function. Considering the above-mentioned challenges, there is a need to train and evaluate the performance of some 2-D CNN classifiers using images that are generated from CWT and

S-transform. The models classify eleven types of arrhythmias and the normal sinus rhythm using MIT-BIH dataset.

1.3 Objectives of the Study

1.3.1 General Objective

The main aim of this study is to find the most suitable 2-D CNN classifier that is capable of detecting cardiac arrhythmias and classifying them into eleven classes and normal rhythm based on electrocardiogram signals (ECGs), as the rest are less common.

1.3.2 Specific Objectives

The specific objectives are:

- To investigate the application of using CWT and S-transform in generating images for arrhythmia's classification.
- To train and evaluate the performance of different 2-D CNN models in classifying joint time frequency representation of ECG signals.
- To find the best set of hyper-parameters for training each model.
- To compare the output resulted from trained models with each other and find the best model that can be used for classifying both CWT and S-transform generated images.

1.4 Research Question/Hypothesis

- How can the use of CWT and S-transform affect the performance of the model's training?
- Which 2-D CNN classifier can give the best performance in classifying arrhythmias?
- What is the best set of hyper-parameter that can be used to train CNN models for classifying arrhythmias?

1.5 Significance of the Thesis

The use of 2-D CNN to classify time-frequency representation of ECG waveforms can be multilateral due to the use of 2-D kernels which could help in providing representative features for time series data. In addition, it has higher flexibility in term of data augmentation. The scaled and shifted versions of the time-localized mother wavelet functions that are used in CWT to create a signal's time-frequency representation could provide excellent time and frequency localization, compared to Fourier Transform which decomposes a signal into infinite length sines and cosines. Also, using S-transform with localizing Gaussian window can improve the performance over the fixed width window utilized in the STFT. The finding

of this study can classify the cardiac arrhythmias into twelve classes, thereby easing the subsequent treatment based on the type of arrhythmia. The use of the proposed automatic technique in this thesis can overcome the drawbacks of the traditional method of classification. This is important in early arrhythmias detection and classification which can be life-threatening.

1.6 Scope and Limitations of the Study

The methodology developed in this thesis for classification of cardiac arrhythmias has been tested on ECG signals. The study included twelve types of arrhythmias, as the rest are less common. The problem in classifying the arrhythmias is that two distinct diseases may have the same effects on normal ECG signals, which can complicate the classification and affect the accuracy. The proposed study will be limited to the available dataset for both training and testing of the models. Collecting data from different sites (hospitals) might improve the result and make it more general. The execution of such is subject to availability of more data. The methodology developed has been implemented entirely on a Matlab 2020a environment.

1.7 Organization of the Thesis

The rest of the thesis has been organized into four chapters. Chapter 2 includes the anatomy of the heart, basic concepts and the science behind ECG signals acquisition, and machine learning techniques used in classifying the different types of arrhythmias. Literature review is included in Chapter 3. In Chapter 4, the cardiac arrhythmias classification models are explained including the pre-processing, models modifications and classification steps. Results and discussions are included in Chapter 5, while Chapter 6 concludes the thesis by giving some useful recommendations and possible future directions.

CHAPTER 2

Theoretical Background

2.1 Anatomy of the Heart

The heart has four chambers: right atrium, left atrium, right ventricle and left ventricle in addition to many atrioventricular and sinoatrial nodes. The left and right atria are the two upper chambers, while the left and right ventricles are the two bottom chambers. Fibrous, non-conductive tissue connects the atria to the ventricles, keeping them separate. The right atrium contracts and drives blood into the right ventricle, expanding the ventricle and increasing its pumping efficiency (contraction). Then after, the right ventricle pumps the blood to the lungs, where it is oxygenated. Similarly, the left atrium and left ventricle work together to push oxygen-rich blood received from the lungs (through the pulmonary veins) to the rest of the body [**26**], as shown in Fig. 4.

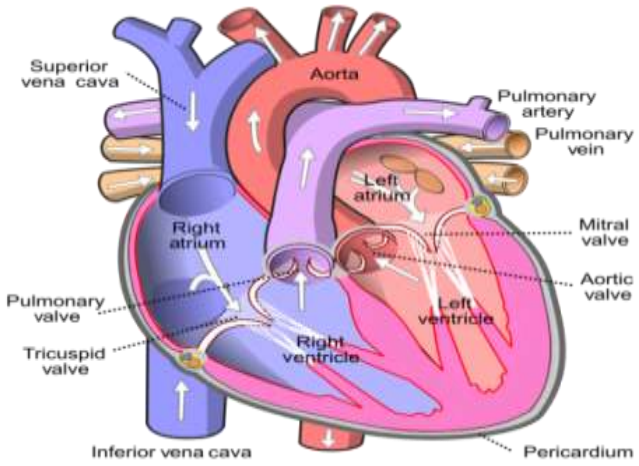


Figure 4 Anatomy of the heart

Adapted from: https://en.wikipedia.org/wiki/Atrium_(heart)#/media/File:Diagram_of_the_human_heart_(cropped).svg

(**Accessed on; June 5th 2021**)

The Sino-atrial (S-A) node in the heart generates regular electrical impulses that travel through the heart's conduction system and cause the myocardium to contract. Depolarization is a process that allows an electrical impulse to go through excitable tissue. A significant ionic current is generated when the cardiac muscles depolarize collectively [**27**]. This current causes a voltage drop as it passes through the resistive bodily tissue. The magnitude of the

voltage drop is big enough for electrodes attached to the skin to detect it. ECGs are recordings of voltage decreases across the skin induced by ionic current flow created by cardiac depolarizations [**28**]. The P-wave is caused by atrial depolarization, which causes the electrical impulse to diffuse throughout the atrial myocardium. Ventricular depolarisation, on the other hand, causes the electrical impulse to diffuse throughout the ventricular myocardium.

2.2 Arrhythmias in ECG signals

Normal sinus rhythm (NSR) is the heart's normal rhythm when there is no irregularity. The heart rate of NSR is usually between 60 and 100 beats per minute. Tachycardia is a rhythm that occurs when the heart rate exceeds 100 beats per minute [**27**]. Bradycardia is a condition in which the heart rate is abnormally slow. When the heart rate is excessively high, the ventricles are not completely filled before contraction, resulting in decreased pumping efficiency. Arrhythmia can be classified into six major types as explained below.

2.2.1 Sinus Node Arrhythmias

The S-A node of the heart causes this form of arrhythmias. In these arrhythmias, the P-wave shape of the ECG is normal, as the electrical impulse is generated by a normal pacemaker (see also Fig. 5). Examples of this arrhythmia include sinus arrhythmia and sinus bradycardia.

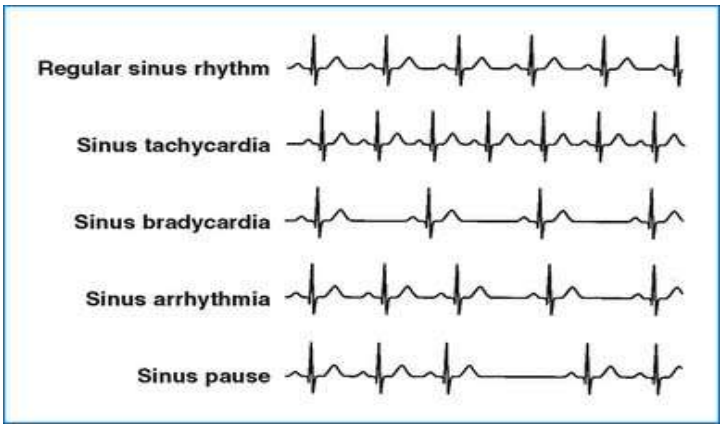


Figure 5 Normal and abnormal rhythms originating in the sinoatrial node

Adapted from: https://thoracickey.com/wp-content/uploads/2016/06/B9781416064435500107_f05-01-9781416064435.jpg (**Accessed on; June 13th 2021**)

2.2.2 Atrial Arrhythmias

The cause of atrial arrhythmias is electrical impulses from outside the S-A node, yet within the atria. There are different types of these arrhythmias. Premature Atrial Contractions (PAC) happens when the irregular P-wave shape is followed by a normal QRS complex and a T-

wave in this arrhythmia. When three or more PACs occur in a row, the rhythm is called atrial tachycardia which is characterized by a rapid heart rate that varies from 160 to 240 beats per minute. The atrial rate in Atrial Flutter is extremely fast, ranging from 240 to 360 beats per minute. In this case, P-waves are abnormally frequent and fast take on the shape of a saw-tooth waveform, which is known as flutter (F) wave. In Atrial Fibrillation, the atrial rate reaches 350 beats per minute. The uncoordinated activation and contraction of separate sections of the atria causes this arrhythmia. Figure 6 graphically depicts the different atrial arrhythmia types.

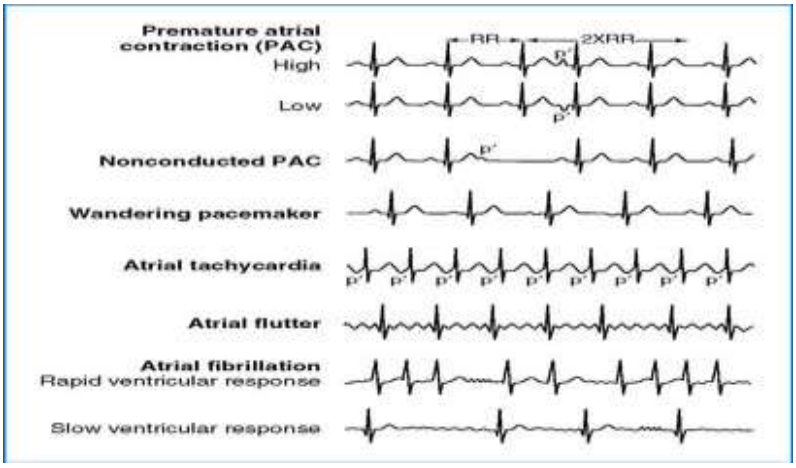


Figure 6 Atrial Arrhythmias types

Adapted from: https://thoracickey.com/wp-content/uploads/2016/06/B9781416064435500107_f05-02-9781416064435.jpg

(**Accessed on; June 13th 2021**)

2.2.3 Junctional Arrhythmias

Junctional arrhythmias are caused by an impulse that originates in the A-V junction and includes the A-V node and its Bundle. Because of these arrhythmias, irregular P wave shape appears. Because depolarization propagates in the opposite direction – from the A-V node to the atria – the polarity of the abnormal P-wave would be opposite to the polarity of the normal sinus P-wave [**27**]. A normal-looking QRS complex arises prematurely in Premature Junctional Escape contraction, but without a preceding P-wave, yet the T-wave morphology is normal (see also Fig. 7).

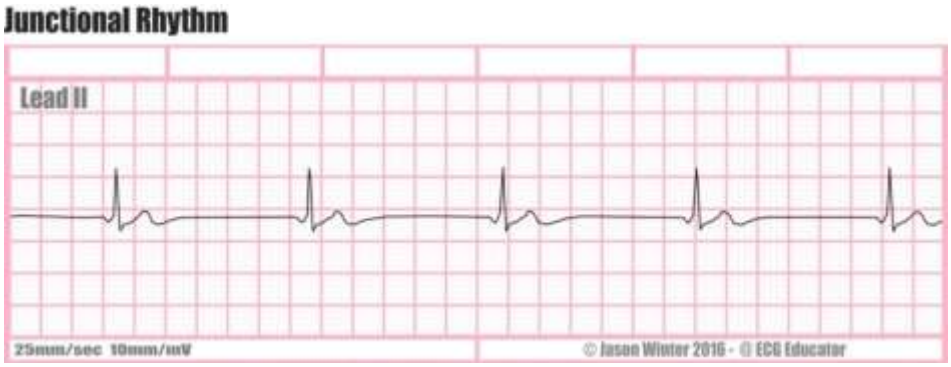


Figure 7 Junctional Arrhythmias

Adapted from: https://i0.wp.com/bossrn.com/wp-content/uploads/2018/06/16-Junctional-Rhythm-STRIP-2.jpg?resize=1080%2C382&ssl=1 (**Accessed on; June 13th 2021**)

2.2.4 Ventricular Arrhythmias

The impulses in this form of arrhythmia originate in the ventricles and go outwardly to the whole of the heart. The QRS-complex is broad and oddly shaped in ventricular arrhythmias. The irregularity in Premature Ventricular Contraction (PVC) is caused by the ventricles. The P-wave morphology maintains its underlying rhythm and occurs at the expected time. PVCs can happen at any point during the heartbeat cycle. Ventricular Tachycardia (VT) has a heart rate of 110 to 250 beats per minute. The QRS complex in VT is excessively wide, unusually shaped, and has a different orientation than the normal QRS complex. Because the high rate may preclude effective ventricular filling and result in a reduction in cardiac output, VT is considered life-threatening. Ventricular Fibrillation is characterized by a fast ventricular rate and an ECG waveform that looks like a saw-tooth (see also Fig. 8).

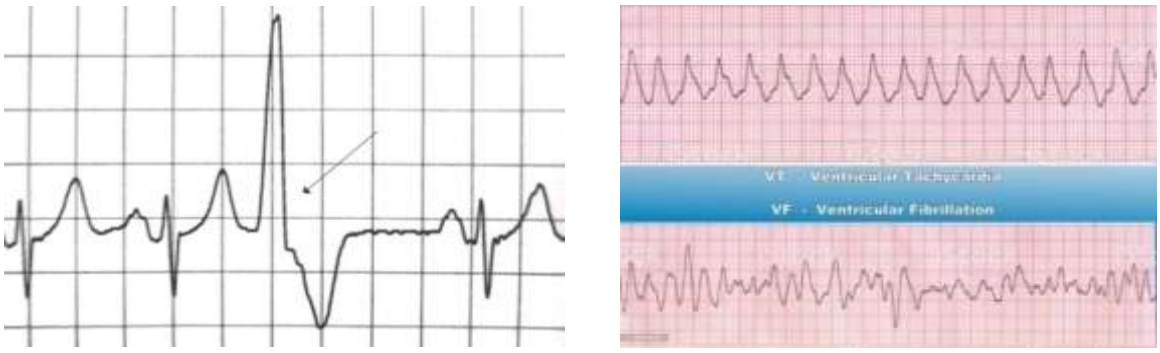


Figure 8 Premature Ventricular Contractions (left), Ventricular Tachycardia and Fibrillation (right)

Adapted from: https://upload.wikimedia.org/wikipedia/commons/a/a5/PVC10.JPG

https://media.istockphoto.com/photos/ventricular-tachycardia-ventricular-fibrillation-picture-id1132930401?s=612x612

(**Accessed on; June 13th 2021**)

2.2.5 Atrio-Ventricular (AV) Blocks

The electrical impulse normally travels via the conduction routes to the ventricles, but the block may delay or prevent the impulse from reaching the rest of the conduction system. When all P-waves are delivered to the ventricles but the PR-interval is prolonged, this is referred to as a first-degree AV block. When some P-waves fail to transmit to the ventricles, second-degree AV blockages result. The rhythm of the P-waves is totally separated from the rhythm of the QRS-complexes in third-degree AV block [**27**]. Figure 9 presents the different AV block types: first-degree AV block, two types of second-degree AV blocks and third-degree AV block.

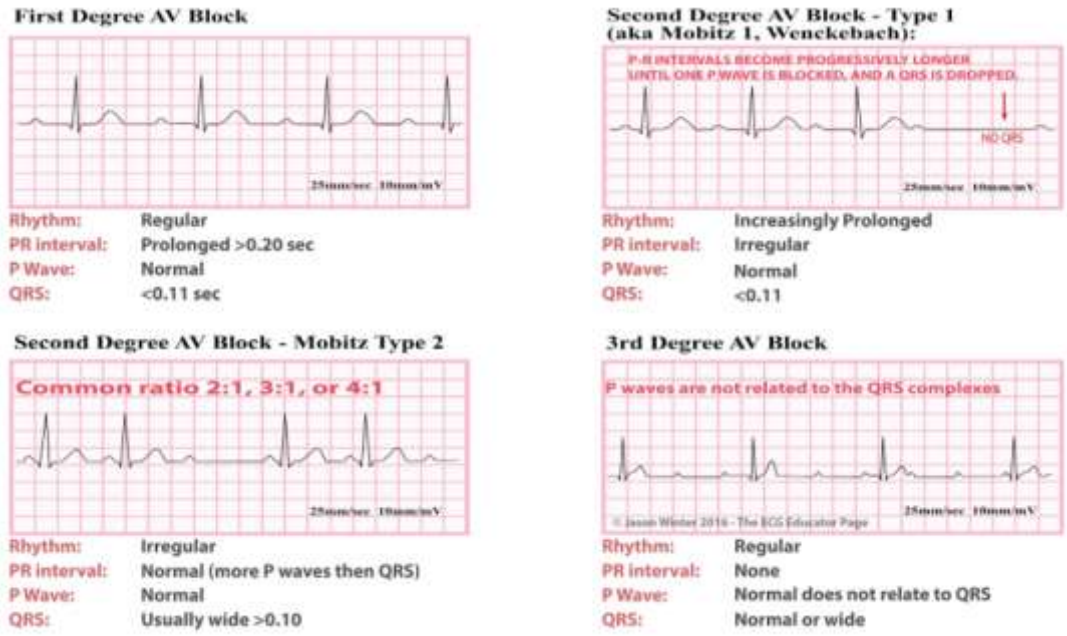


Figure 9 AV blocks types.

Adapted from: https://img.grepmed.com/uploads/751/ecgeducator-wenckebach-heartblock-medstudent-cardiology-original.jpeg (**Accessed on; June 13th 2021**)

2.2.6 Bundle Branch Blocks

The transmission of the impulse from the AV-node to the entire conduction system is halted by a bundle branch block. The beat of the bundle branch block is divided into two types: Left bundle branch block beat (LBBB) and Right bundle branch block beat (RBBB), as also been depicted in Fig. 10. The left bundle branch in LBBB prevents the A-V node's electrical impulses from depolarizing the left ventricular myocardium in the regular way. The electrical impulse from the AV node is unable to propagate to the conduction network to depolarize the right ventricular myocardium when the right bundle branch is blocked.

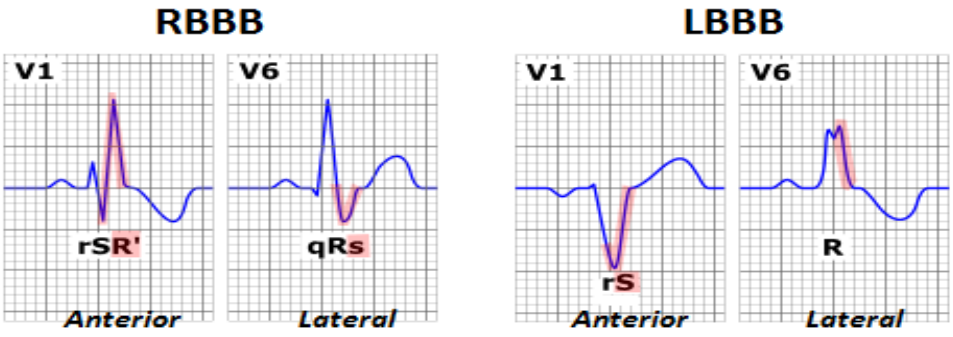


Figure 10 Bundle Branch blocks

Adapted from https://upload.wikimedia.org/wikipedia/commons/3/32/Left_and_right_bundle_branch_block.png

(**Accessed on; June 13th 2021**)

2.3 Arrhythmias Diagnosis

Electrocardiogram (ECG) is a popular test used to monitor the electrical activities of the heart. It is depicted with a waveform signal that shows the heart's electrical activity in rhythm. The machine's wires are connected to different areas of the body: chest, legs and arms. They are characterized as periodic signals because they are made up of a series of waves (patterns recorded on an ECG owing to needle fluctuation) that repeat in time [**29**].

Another test is Holter Monitoring (see Fig. 11), which is a continuous ECG recording of the heart for 24 to 48 hours as the patient goes about their daily activities. Wires are linked from a portable ECG device to the patient's chest for this monitoring [**30**].

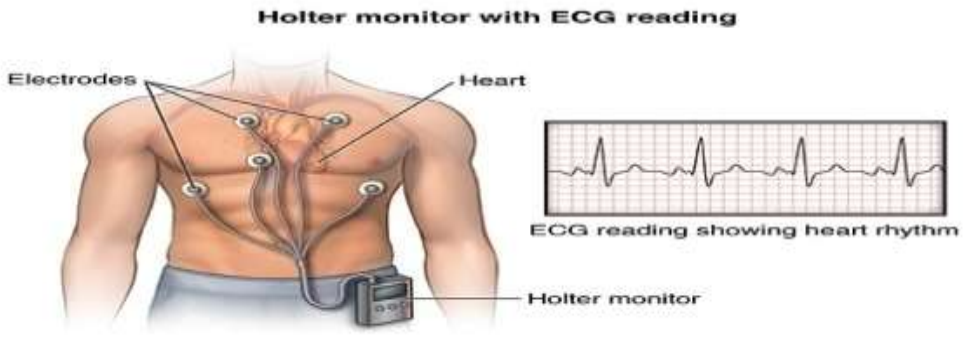


Figure 11 Holter monitor

Adapted from https://www.hopkinsmedicine.org/-/media/images/health/2_-treatment/cardiovascular/holter-monitor.ashx

(**Accessed on; June 20th 2021**)

Exercise stress testing is another useful test for identifying arrhythmia. A 12-lead ECG is attached to a patient during exercise, usually on a treadmill or a bicycle, in this test as can be seen in Fig. 12 [**31**].

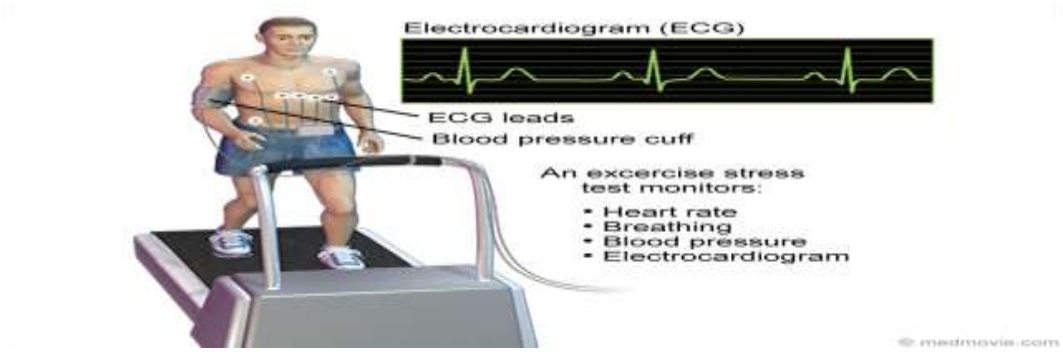


Figure 12 Exercise stress test

Adapted from https://medmovie.com/_uploads/ahaw_0252i_640x480.png

(**Accessed on; June 20th 2021**)

2.3.1 ECG Signal

The heart contracts in a rhythmical way in order to pump blood all over the body. The muscle's contraction begins at the atrial sine node (SA, natural pacemaker), and spreads through the muscle [**32**]. This activity generates electrical currents on the body's surface, exciting the electrical potential of the skin surface to change. These signals can be captured by using electrodes and proper equipment.

Electrocardiogram signal (ECG) illustrates the heart's electrical activity while it operates, via measuring the heart voltage potential differential. This electrical wave is a result of depolarization and re-polarization of Sodium (Na+) and Potassium (k-) ions in the cell [**33**].

The typical ECG waveform has four main elements per each cycle:

- P-wave: it is the initial wave of the ECG cycle; which represents the atrial contraction named as atrial depolarization.
- QRS complex: it is a combination of three deflections and illustrates the contraction of the ventricles; Q-wave describes the depolarization of the inter-ventricular septum, the R-wave is the depolarization of the main mass of the ventricles, and the S-wave is the depolarization at the base of the heart.
- T-wave: describes the ventricle's re-polarization.
- U-wave: describes papillary muscle repolarization.

Other important parameters:

- PR interval: it consists of the P wave and the PR segment. This interval records the time needed by the wave to propagate from the sinus node (SA node) through the myocardium electrical conduction system.

- PR segment: it begins with the P wave and ends at the QRS complex.
- ST segment: the interval from the end of the QRS complex to the start of the T wave.
- QT interval: it is the duration between the beginning of the QRS complex and the end of the T wave, it describes the ventricular depolarization and repolarization [**34**].

Figure 13 presents the different elements of an ECG waveform.

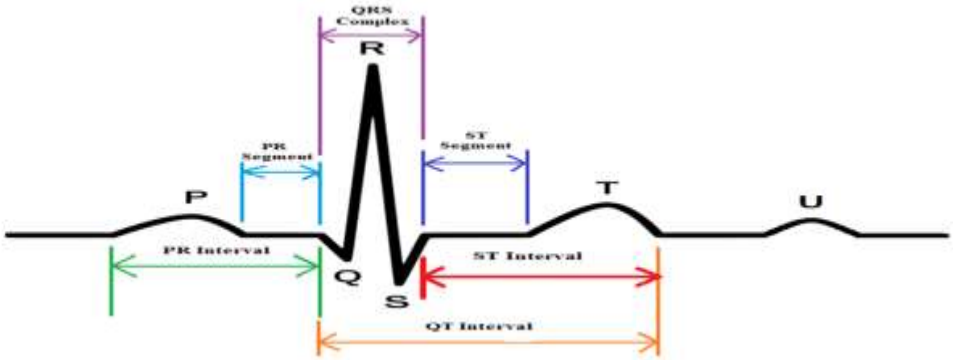


Figure 13 Elements of ECG waveform

Adapted from:

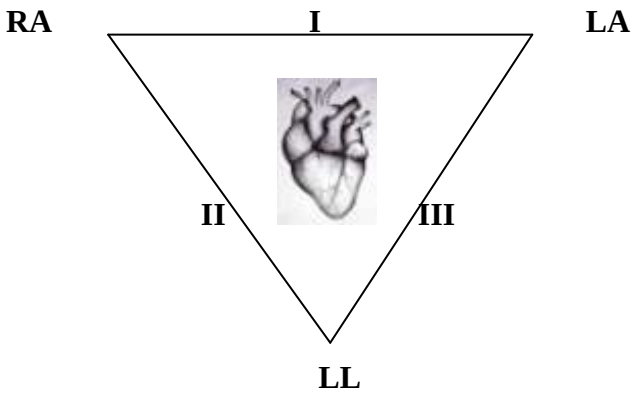
https://www.researchgate.net/profile/RajuSinha/publication/287200946/figure/fig3/AS:329418988900358@1455551015911/Schematic-representation-of-normal-ECG-waveform.png **(Accessed on; Feb 9th 2021)**

2.3.2 ECG Leads

Three bipolar leads, three modified unipolar leads, and six chest (precordial) leads make up the typical ECG. A lead is a set of electrodes (+ve & -ve) attached to an ECG recorder and positioned on the body at certain anatomical regions [**35**]. Recording the potential difference between two points using two electrodes is done by using bipolar leads (Einthoven's Leads), and the unipolar leads record the electrical potential at a specific point(has a single positive electrode and a reference point).

2.3.2.1 Einthoven's Leads

The electrical potential differences between two electrodes are compared in Leads I, II, and III. Because its exploration electrode is on the left, lead I records potentials between the left and right arm. This lead observes the heart from the left. Lead II records potentials between the right arm and the left leg, with the leg electrode being the exploring electrode; It looks at the heart from a 60-degree angle. Lead III looks at the heart from a 120-degree angle and records potentials between the left arm and leg as also been depicted in Fig. 14 [**36**].



Lead	Positive Electrode	Negative Electrode
I	LA	RA
II	LL	RA
III	LL	LA

Figure 14 The three Standard Limb Leads

2.3.2.2 Goldberger's Leads

They are unipolar limb leads and the exploring electrode (connected to positive terminal) in these leads is compared to a reference electrode that is based on an average of the other two limb electrodes. In a VR, the exploring electrode is the right arm, while the reference electrode is made up of the average of the left arm and left leg. The left arm electrode is exploring in a VL, and the lead is looking at the heart from a -30 degree angle. The positive terminal of lead a VF is attached to the left leg as also been depicted in Fig. 15 [**28**].

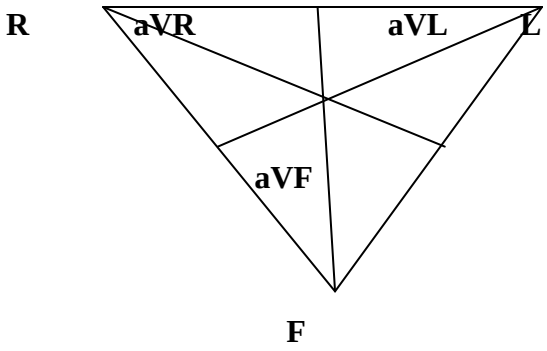


Figure 15 Limb unipolar leads

2.3.2.3 Wilson Leads

Wilson leads (V1–V6) are unipolar chest leads that are positioned in a roughly horizontal plane on the left side of the thorax as depicted in Fig. 16. By connecting the three standard limb leads, the indifferent electrode is formed. They give a 3D view of the integral vector when used in conjunction with the unipolar limb leads in the frontal plane. The three limb

leads are joined to form an indifferent electrode with high resistances in order to make recordings with the chest leads (different electrode). The chest leads are primarily used to detect potential backwards-directed vectors. In the frontal plane, these vectors are barely discernible [**27**]. The QRS vectors recorded by leads V1–V3 are usually negative, whereas those identified by V5 and V6 are usually positive, because the mean QRS vector is normally directed downwards and towards the left back region [**28**]. Because the chest electrode on leads V1 and V2 is closer to the base of the heart, which is the direction of electronegativity for most of the ventricular depolarization process, QRS is negative. Because the chest electrode in these leads is closer to the heart apex, which is the direction of electropositivity during most depolarization, QRS is positive in leads V4, V5, and V6 [**35**].

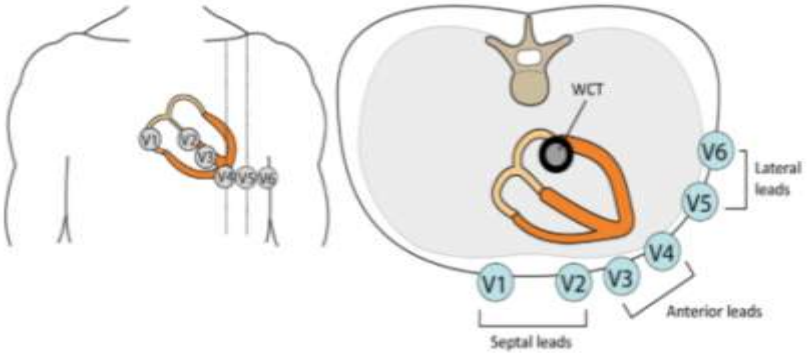


Figure 16 The chest leads (precordial)

Adapted from: https://ecgwaves.com/topic/ekg-ecg-leads-electrodes-systems-limb-chest-precordial/

(**Accessed on; June 6th 2021**)

2.4 Artificial Intelligence and Machine Learning

Artificial intelligence (AI) is a broad field of computer science that focuses on creating intelligent machines that can accomplish activities that would normally need human intelligence [**37**]. Artificial Intelligence has made significant progress in closing the gap between human and computer capabilities. Researchers work on a variety of areas in the field to achieve incredible results [**16**]. The domain of Machine Learning is one of several such topics. Machine learning is a method of artificial intelligence in which a computer algorithm (a collection of rules and procedures) is created to assess and predict data fed into the system. There are two types of existing models for automatic diagnosis of cardiac arrhythmias: traditional approaches and deep learning methods. Traditional machine learning (ML) methods include two stages: first, specialists must create meaningful features or extract features using signal processing techniques, and then these features must be used to build ML

classifiers [**38**]. The second method is to employ end-to-end deep learning algorithms that do not necessitate explicit feature extraction. Figure 17 demonstrates a comparison between traditional methods and deep learning methods.

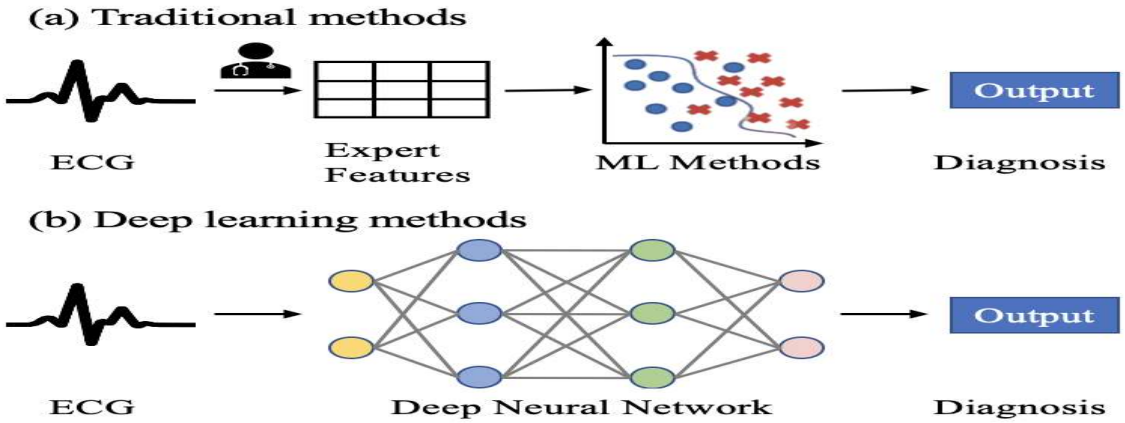


Figure 17 Comparison between automatic methods of ECG signals classification

Adapted from:

https://www.researchgate.net/publication/344780112/figure/fig1/AS:948916233175040@1603250668022/Comparison-of-existing-models-for-automatic-diagnosis-of-ECG-abnormalities-a-two-stage.ppm (**Accessed on; June 20th 2021**)

2.4.1 Artificial Neural Networks

Artificial Neural Networks (ANNs) is a deep learning technique which has history that stretches back in the nineteenth century, when Warren McCulloch and Walter Pitts [**39**] created a computational model for neural networks in 1943. Around 1975, Werbos's was able to develop back-propagation algorithm which enabled practical training of multi-layer networks [**40**]. Recently, extensive studies have been carried out using ANNs.

ANNs were basically developed as mathematical theories of the information-processing activity of biological nerve cells [**41**]. The electrical signal in a biological neuron is carried from the dendrites to the cell body, and then a single output or electrical signal is passed to neighbouring neurons via the axon. Similarly, we try to replicate this behaviour in an artificial neuron (perceptron) with inputs and outputs (see also Fig. 18). ANNs can have single or multiple layers [**41**], these layers consist of neurons (processing units) connected together by a set of adjustable weights that allow signals to travel parallel and consecutively through the network [**42**] [**43**].

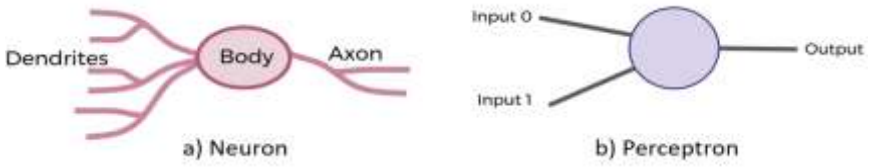


Figure 18 Representation of a) biological neuron and b) artificial neuron

Adapted from: https://punndeeplearningblog.com/wp-content/uploads/2021/04/Capturse-768x177.png (Accessed on; Feb 26th 2021)

Generally, an ANN can be divided into three layers of neurons: input (receives information), hidden (responsible for extracting patterns, perform most of the internal processing), and output (produces final network outputs) [**44**] (see also Fig. 19).

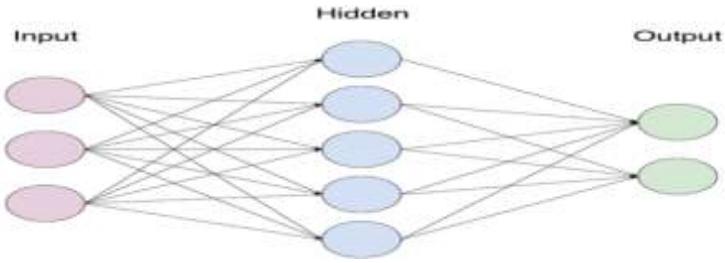


Figure 19 Basic artificial neural network components

Adapted from: https://cdn-images-1.medium.com/proxy/1*RGV6Bb3ChmVWsA8Q6Qth6Q.png

(**Accessed on; Nov 5th 2020**)

Each neuron within the network has a weighted input, transfer function and a single output. The neuron receives the weighted input and activates the output produced by passing the activation signal through a transfer function. The neural network's behaviour is determined by the neuron's transfer functions, the learning rule and the architecture of the network as also depicted on Fig. 20.

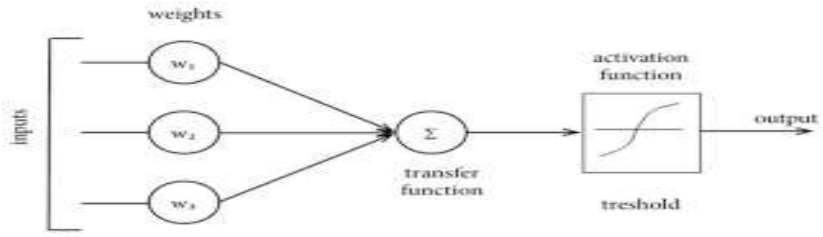


Figure 20 Single-layer perceptron

Adapted from: https://www.google.com/search?q=artificial+neurons+component&hl=en US&source=lnms&tbm=isch&sa=X&ved=2ahUKEwj5habs1OrsAhWSAGMBHeTFA_cQ_AUoAXoECBUQAw&biw=1366&bih=662 (**Accessed on; Nov 5th 2020**)

Mathematically, neural networks are represented as:

$$y = f \sum_{i=1}^{n} (w_i \, x_i)$$

where x_i represents the inputs, w_i are the different weights of the inputs which passes through the activation function f to give us the response$y$.

The selection of suitable training methods [**45**] or using pre-processing techniques [**46**], the use of balanced datasets [**47**] and adequacy [**48**] of datasets are important factors to achieve satisfactory performance of ANNs. The advantages of ANN compared to traditional techniques include fast and simple operations, ability to operate with noisy information, ability to learn inductively from training data and process non-linear functionality to deal with real-world data [**49**].

2.4.1.1 Architecture of ANN

Architectures of ANNs are commonly classified as:

- Feed-forward neural networks, which can be single-layer or multi-layer network.
- Feed-back (recurrent neural networks) [**50**].

In the feed-forward networks, there is no any feedback; the information flow is unidirectional from input layer, through hidden layers to the output layer. On the other hand, a recurrent network involves dynamic information processing having at least one feedback loop, using outputs as feedback inputs [**44**] [**51**]. The two architectures have been depicted in Fig. 21.

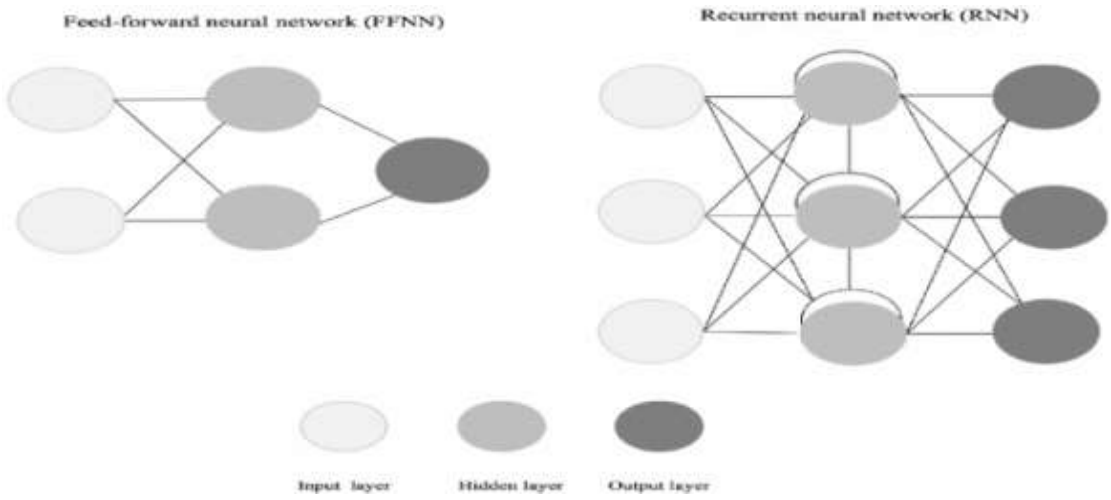


Figure 21. Feed-forward and recurrent neural network models

Adapted from:https://journals.plos.org/plosone/article/file?id=10.1371/journal.pone.0212356.g001&type=large

(**Accessed on; Nov 3rd 2020**)

- **Activation Function**

The Activation Function is a critical element of Neural Networks. It's employed as a decision maker in a network's output layer to map value to the final output, with the intended range of 0 to 1. In a deep neural network, the activation function introduces non-linearity to the network via the hidden layer. It can be linear or nonlinear [**52**]. These are used in the hidden and output layers to assist the network in discovering hidden data patterns. The following are the most commonly used activation functions in neural networks [**53**]:

- **Sigmoid function:** its value spans between 0 and 1, the sigmoid function is most commonly utilized in the output layer for binary classification.
- **Tanh activation function:** is superior to sigmoid activation function because it provides more non-linearity. Its value varies from -1 to 1. It is, in fact, the scaled version of the sigmoid function. As a result, we can easily swap sigmoid with tanh and still achieve good results.
- **ReLU:** is the most often utilized activation function since it provides better gradients even for large values while not reducing the model's learning rate.
- **Leaky ReLU:** is an extension of the ReLU function in which the gradient or slope is not zero for negative values. However, because performance is rarely much impacted in practice, ReLU is the most often utilized activation function.

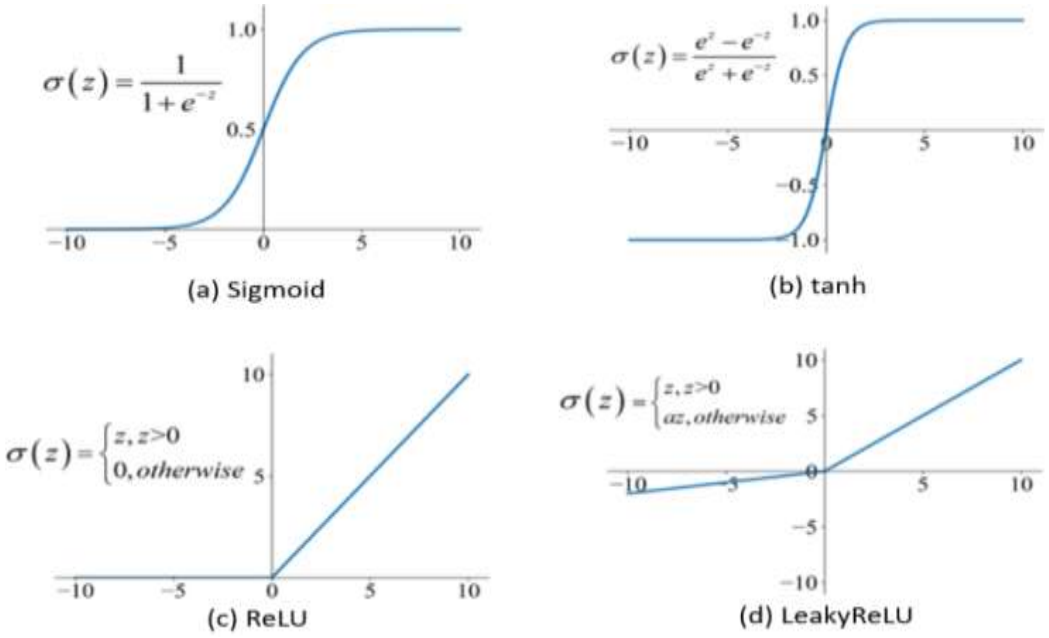


Figure 22 Activation functions that are used in neural networks: a) Sigmoid, b) Tanh, c)ReLU, and d) LeakyReLU

Adapted from: https://punndeeplearningblog.com/wp-content/uploads/2021/04/af-768x545.png (**Accessed on; Feb 26th 2021**)

2.4.2 Convolutional Neural Networks

A convolutional neural network (CNN) is a type of neural network that is used to analyze input using a specified grid-like architecture. CNNs can take an input image, give relevance (learnable weights and biases) to various aspects/objects in the image, and distinguish between them. In comparison to other classification algorithms, CNNs require substantially less pre-processing [**54**]. The CNN goal is to compress the images into a format that is easier to process while preserving elements that are important for obtaining a decent prediction. The architecture of CNN is inspired by the organization of the Visual Cortex and is akin to the connectivity pattern of neurons in the human brain. Individual neurons can only respond to stimuli in a small area of the visual field called the Receptive Field. A number of similar fields can be stacked on top of each other to span the full visual field. The following layers make up the deep CNN [**17**]: Convolution, ReLU (Rectified Linear Units), Pooling, Fully Connected, and Softmax. The basic components of CNNs are depicted in Fig. 23 including the different layers.

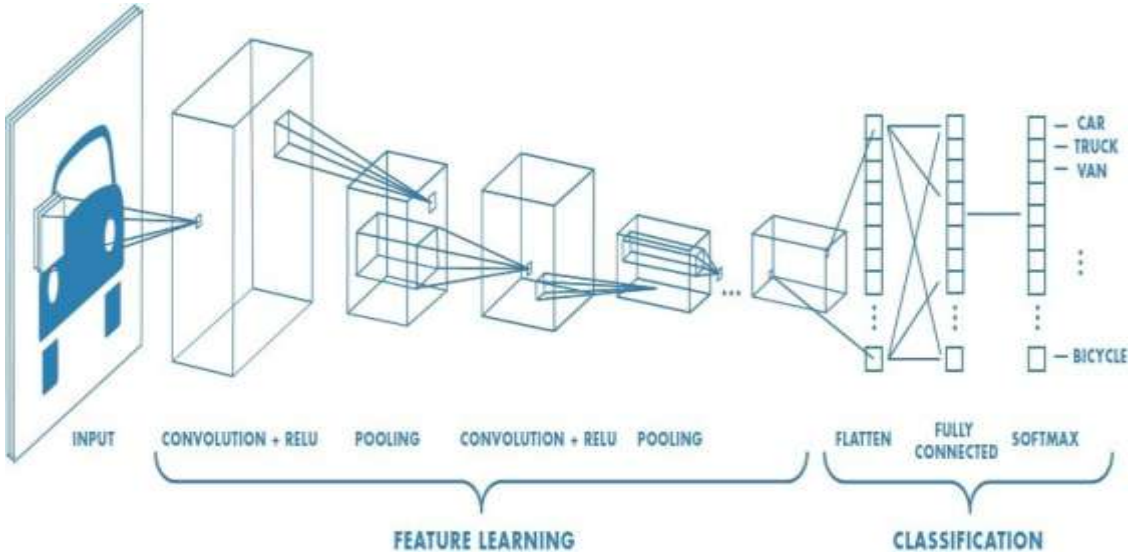


Figure 23 Basic components of convolutional neural networks

Adapted from :https://miro.medium.com/max/2000/1*vkQ0hXDaQv57sALXAJquxA.jpeg **(Accessed on; Mar 15th 2021)**

2.4.2.1 Convolution Layers

The goal of the convolution operation is to extract high-level characteristics from the input image, such as edges. CNN does not have to use just one convolutional layer; the first ConvLayer is in charge of capturing low-level characteristics like edges. The architecture adjusts to the high-level characteristics as well with the addition of layers, and this will help the network to understand all of the images in the dataset in the same way that we do [**54**]. The Kernel/Filter is the component that performs the convolution operation in a

Convolutional Layer. With a given Stride Value, the filter progresses to the right until it parses the entire width. Moving on, it uses the same Stride Value to return to the beginning (left) of the image and repeats the process until the full image has been traversed. Every time we perform a matrix multiplication operation between the filter and the portion of the image over which the kernel is hovering. A single value is obtained by multiplying the filter with the input array once. Because the filter is applied to the input array several times, the outcome is a 2D array of output values that indicate input filtering. This operation's 2D output array is referred to as a "feature map" [**55**]. The filter that is used to create a feature map from a 2D image input is depicted in Fig. 24.

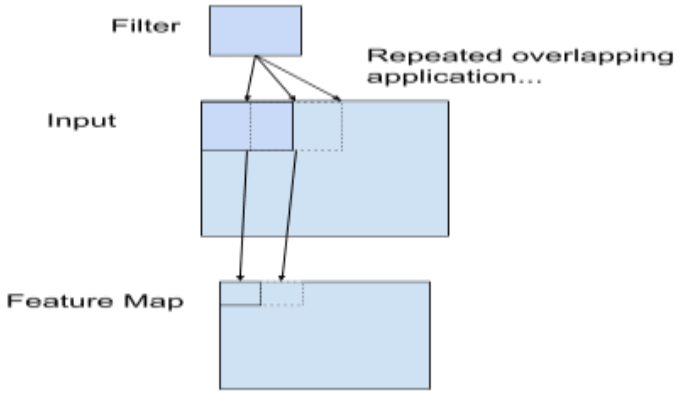


Figure 24 Filter Used to Create a Feature Map from a Two-Dimensional Input

Adapted from: *https://machinelearningmastery.com/wp-content/uploads/2019/01/Example-of-a-Filter-Applied-to-a-Two-Dimensional-input-to-create-a-Feature-Map.png (***Accessed on; Mar 17th 2021)**

2.4.2.2 Rectified Linear Units (ReLU) Layers

ReLU layer is used to filter information as it travels over the network represented by the function:

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

where x is the input. It performs an element-wise operation on the input and, if the value is negative, it sets it to zero. If the result is positive, it will simply be transferred through its identity [**56**]. A rough schematic of the ReLU layers is presented in Fig. 25.

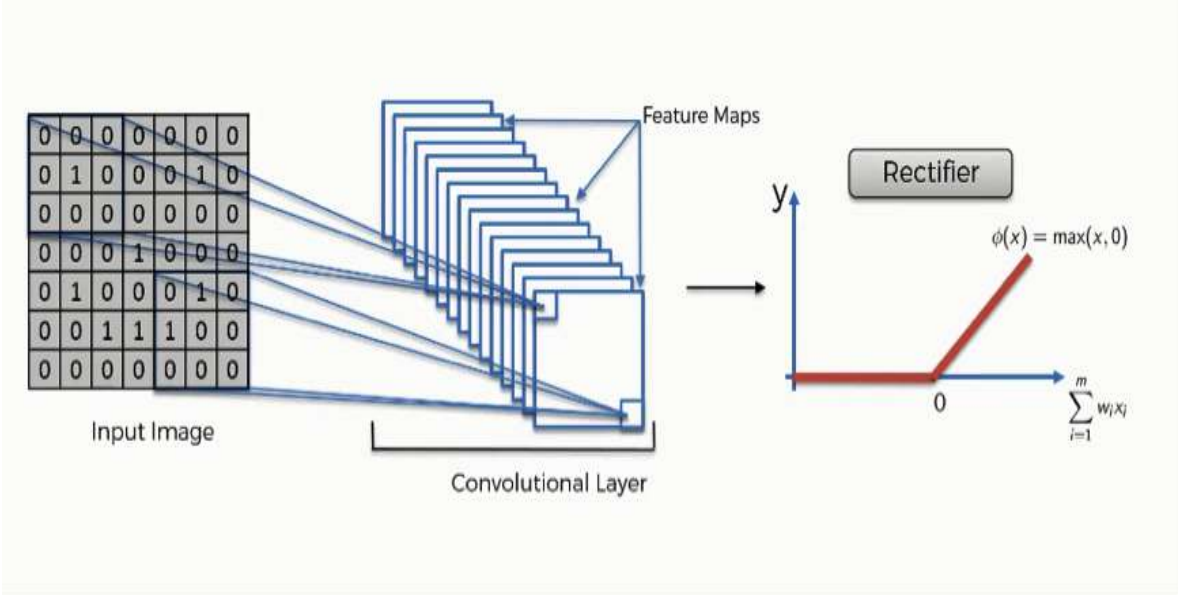


Figure 25 Rectified Linear Units in CNN

Adapted from: *https://sds-platform-private.s3-us-east-2.amazonaws.com/uploads/71_blog_image_1.png* **(Accessed on; Mar 17th 2021)**

2.4.2.3 Pooling layers

The convolved feature's spatial size is reduced according to the pooling layer. Through dimensionality reduction, the computer power required to process the data is reduced. It's also beneficial for extracting different features, which helps keep the model's training process running smoothly. The commonly used pooling techniques are: maximum pooling and average pooling. The maximum value from the portion of the image covered by the Kernel is returned by Max Pooling. Average Pooling, on the other hand, returns the average of all the values from the Kernel's section of the image as also depicted in Fig. 26.

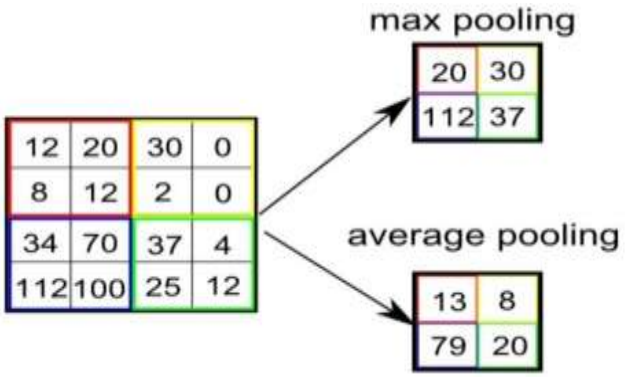


Figure 26 Example of max pooling and average pooling in CNN

Adapted from: https://editor.analyticsvidhya.com/uploads/597371_KQIEqhxzICU7thjaQBfPBQ.png **(Accessed on; Mar 17th 2021)**

2.4.2.4 Fully Connected (FC) Layers

The output of the final Pooling or Convolutional Layer flattened and then fed as input to the fully connected layer. These layers produce a 2D matrix, which can be flattened by unrolling all of its values into a vector as also depicted in Fig. 27 [**54**].

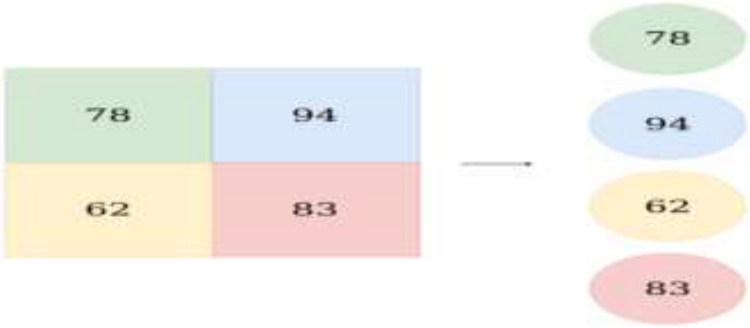


Figure 27 Example of flattening before feeding to FC

Adapted from: https://miro.medium.com/max/2000/1*xY34n_U-IQdCXTcd2zXYQw.png **(Accessed on; Mar 20th 2021)**

This flattened vector is then passed to a few fully connected layers that conduct the same mathematical operations as ANNs (see also Fig. 28).

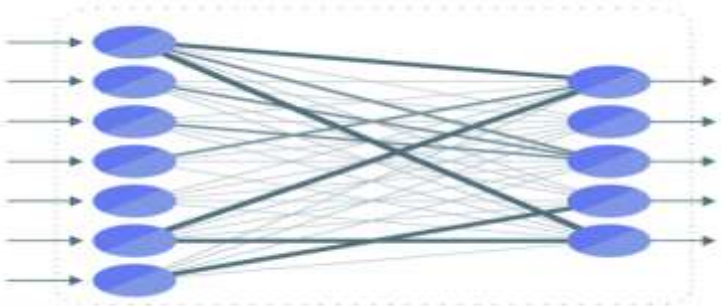


Figure 28 Fully connected layers in CNN

Adapted from: *https://miro.medium.com/max/439/1*sVvC9YwPFD5RJ9xgxrYHPw.png* **(Accessed on; Mar 20th 2021)**

2.4.2.5 Softmax

After passing through the fully connected layers, the final layer employs the softmax activation function to determine the likelihood that the input belongs to one of several classes. The softmax function converts a vector of real values to a vector of real values that add up to one. The softmax turns the input values, which might be positive, negative, zero, or higher than one, into values between 0 and 1, allowing them to be understood as probabilities [**57**]. The softmax function is defined by the formula [**58**]:

$$\sigma(z)_i = \frac{e^{z_i}}{\sum_{j=1}^{k} e^{z_j}}$$

It normalizes each component z_i of the input vector z by dividing by the sum of all these exponentials; this normalization assures that the sum of the output vector's components $\sigma(z)$ is 1 as also demonstrated in Fig. 29.

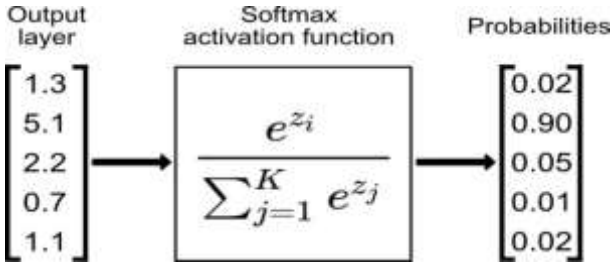


Figure 29 Example of softmax function

Adapted from:https://miro.medium.com/max/700/1*ReYpdIZ3ZSAPb2W8cJpkBg.jpeg (**Accessed on; Mar 20th 2021**)

2.4.3 Calculating Parameters for ANNs

2.4.3.1 Loss Function

It's a function that determines how well a model performs for any given set of data. When the input vectors are transmitted forward through the network, the actual network outputs y are compared to the desired or targeted outputs ŷ, and the difference is computed as the cost function, which is the error. The goal of this function is to inform the model about any network issues that need to be fixed. The two primary forms of cost functions in a network are mean squared error (MSE) and cross-entropy [**52**].

Mean squared error (MSE) is represented mathematically as:

$$MSE = \frac{1}{2}(y - \hat{y})^2$$

For cross-entropy loss (log loss), the most common application is classification. The kind chosen depends on the classification problem; binary classification employs sigmoid output to provide a probability prediction result of either 0 or 1, whereas categorical prediction uses softmax for multi-class classification [**52**].

The mathematical representations of cross-entropy loss (l) for binary and multiclass classification are given respectively by:

$$l = -(y.\log(\hat{y}) + (1 - y)\log(1 - \hat{y}))$$

$$l = -\sum_i y_i \log \hat{y}_i$$

One can calculate the cost function after forming a hypothesis with starting parameters. In order to lower the cost function, we use the Gradient descent technique to change the parameters over the given data [**59**].

2.4.3.2 Gradient-based Optimization

It is important the process of achieving minimum loss value (convergence to global minima) be fast and simple. The optimization procedure is used to train neural networks. The goal is to use optimization to reduce the loss function's inputs, which is a brute force technique. When the cost function has been set to those neurons, gradient descent can be used to update the weight at each neuron. Gradient descent (steepest descent) is an iterative optimization approach for determining a function's local minimum. The concept of gradient descent was proposed in 1847. By using gradient descent to discover a function's local minimum, we take steps proportional to the negative of the function's gradient (move away from the gradient) at the present location (see also Fig. 30). Gradient ascent is the process of approaching a local maximum of the function by taking steps proportional to the positive of the gradient (going towards the gradient) [**60**].

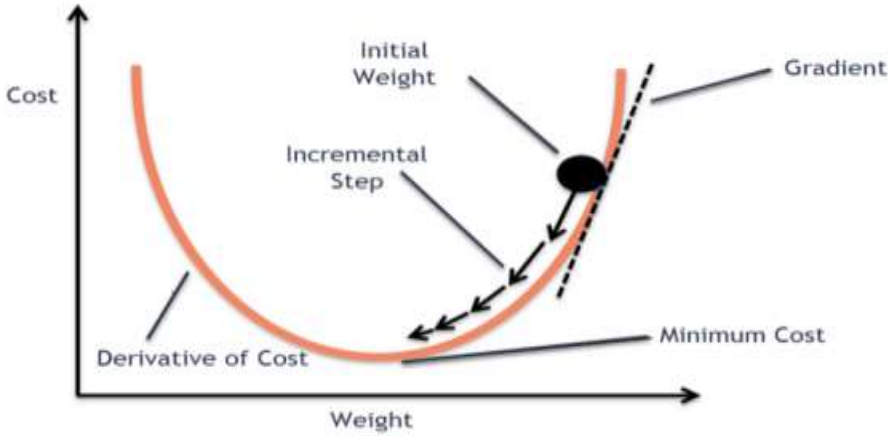


Figure 30 Gradient Descent approach

Adapted from : https://editor.analyticsvidhya.com/uploads/631731_P7z2BKhd0R-9uyn9ThDasA.png **(Accessed on; July 22nd 2021)**

The mathematical form for gradient descent is given by:

$$\theta_{y+1} = \theta_y - \epsilon \nabla_\theta f(\theta)$$

where θ is the weight, ϵ is the learning rate, and $\nabla_\theta f(\theta)$ is the partial derivatives of the loss function with respect to the weight θ [**61**].

Based on the amount of data utilized to calculate gradients, there are three main types of gradient descent: Batch gradient descent, stochastic gradient descent, and mini-batch gradient descent [**62**]. Each neuron's weight is updated utilizing stochastic gradient descent. Its rapid updates can create errors at times, although they are usually faster than batch gradient descent. After all of the neurons have been examined, batch gradient descent updates them all at once. However, owing of its processing requirements, it necessitates a large quantity of memory. Mini-batch gradient descent is a deep learning technique that combines batch and stochastic gradient descent.

2.4.3.3 Back-propagation

ANNs gather information and learn by detecting patterns and relationships in the data through experience. It learns by optimizing inner connections between units aiming to minimize errors in the predictions, thus reach a desired level of accuracy. When an input vector is fed into the network to produce an actual output, this output is compared with the desired output. If there is no difference between them, then there is no need for further training. Otherwise, the output is back-propagated and the weights are updated using gradient descent to reduce the differences between the two outputs [**62**]. Once the model has been trained and tested, then new information can be inputted into the model [**43**].

Back-propagation indicates how an ANN is trained or 'learn' based on data. It uses an iterative process involving six steps [**41**]:

(i) Single data is passed into the input layer; the produced output then is passed to the hidden layer and multiplied by the first group of weights matrix.

(ii) The incoming signals are summed, transformed to output and passed to second set of connection weights.

(iii) Then the network output is produced through summing and transforming the incoming signals.

(iv) Network's output value is subtracted from known value for this specific case, and error value is passed backward through the network.

(v) The next step is adjusting the connection weights in proportion to their error contribution.

(vi) Saving the modified connection weights for next cycle, and next case input set queued for next cycle.

Training (learning) methods in ANN are classified into three types: supervised, unsupervised and reinforced learning. When every input pattern is used to train the network and it is

associated with an output pattern, it is named supervised learning. In this method, the error in computed and desired outputs can be used to improve the performance of the model. In unsupervised learning, the network learns by itself without any previous knowledge of desired output and by discovering and adapting to features of the input patterns. In reinforcement learning, the network is provided with feedback on its computation performance without presenting the desired output [**51**].

2.4.3.4 Training and Hyper-parameters

After developing the deep learning model's main architecture, there are still numerous decisions to be taken that will affect its performance. The learning algorithm's behaviour can be controlled using a collection of settings known as hyper-parameters. They govern many parts of the learning algorithm and can have a significant impact on the model's performance. Before beginning the training procedure, the hyper-parameters must be defined [**63**].

Training parameters such as learning rate, batch size, and number of epochs are common hyper-parameters. Hyper-parameters include architectural characteristics such as the number of layers and layer size. Additional parameters introduced by CNNs include the number of filters, filter size, padding, and stride. In order to enhance the utility of the learning process, several hyper-parameter combinations should be evaluated [**64**].

2.4.3.5 Epoch and Batches

Hyper-parameters include both Epochs and Batches. The training set is divided into small batches (mini-batches) as back-propagation-based training algorithms are computationally expensive to calculate for each sample. The amount of samples processed before the model is changed is referred to as batch size. In one training phase, the mean error of the samples in a single batch is used to do back-propagation [**65**]. One epoch is completed when all of the batches that make up the train sets have passed through the network. Epochs are used to train a neural network in an iterative process, and many epochs are usually necessary for good convergence [**64**].

2.5 Time Domain Analysis

A time domain analysis is a time-based analysis of physical signals, mathematical functions, or time series of economic or environmental data. The signal or function's value is also understood for all real numbers at multiple separate instances in the time domain. A time-domain graph can also demonstrate how a signal evolves over time [**66**].

2.6 Frequency Domain Representation

The examination of mathematical functions or signals in the frequency domain, rather than in the time domain, is referred to as frequency domain analysis. To put it another way, a time-domain graph depicts how a signal varies over time, whereas a frequency-domain graph depicts how much of the signal is contained inside each frequency band throughout a variety of frequencies. The phase shift that must be applied to each sinusoid in order to recombine the frequency components and recover the original time signal can also be included in a frequency-domain representation [**67**]. One of the primary benefits of utilizing a frequency-domain representation of a problem is that it makes mathematical analysis easier. A pair of mathematical operators known as transforms can be used to translate a given function or signal between the time and frequency domains. The Fourier transform, for example, transforms a time domain function into a sum or integral of sine waves of various frequencies, each representing a frequency component.

2.6.1 Fourier Transform

Joseph Fourier originally published the Fourier Transform (FT) in 1822 [**68**]. It transforms the time domain of a mathematical function to the frequency domain. This allows us to discover previously unknown properties of the function [**69**]. Figure 31 shows the recovery of signals using the FT.

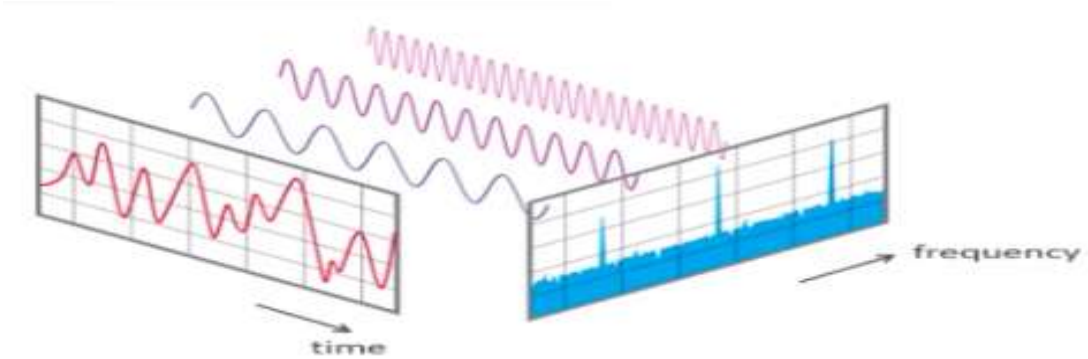


Figure 31 How Fourier Transform recover the original signal. Y-axis is amplitude

Adapted from: https://miro.medium.com/max/471/1*uL4gqMutokf5r-M8P7bG7w.png (**Accessed on; Jan 15th 2021**)

The FT is a method of describing the overall regularity of a periodic signal as well as the related concept of the frequency scale. The FT is based on sinusoidal basis functions and only provides frequency resolution, not time resolution. The FT is a mathematical transformation that decomposes functions depending on space or time into functions depending on spatial or

temporal frequency [**70**]. It is an approach for converting a time function, x(t), to a frequency function, X(ω) as follows [**71**]:

$$X(\omega) = \int_{-\infty}^{+\infty} x(t)\ e^{-j\omega t}\mathrm{dt} \qquad (1)$$

In principle, it is possible to completely recover a time series from its FT using the Inverse FT:

$$x(t) = \frac{1}{2\pi}\int_{-\infty}^{+\infty} X(\omega)\ e^{j\omega t}\mathrm{d}\omega \qquad (2)$$

Although the FT of the full time series contains information on the spectral components of a time series, this information is insufficient for a wide class of practical applications. Because the investigated signals might be locally regular without being globally regular, the FT allows us to study their global regularity and doesn't describe or evaluate the local regularity. The FT doesn't provide information regarding the frequency evolution over time [**72**].

2.7 The Joint Time-frequency Representation (JTFR)

Lately, researchers have become more interested in studying the limitations of the FT. When using the FT to analyze the spectrum of a given signal, it measures the frequency components of the signal but doesn't give the temporal information of the time varying signal [**73**] [**74**]. Also, the FT is not able to distinguish between the same frequencies of vibrations at two different time intervals; as it shows this information as a single peak in the FT plots [**75**].

The joint time-frequency representation (JTFR) of signals is a notable tool that has been developed to analyze the spectral components and the temporal information of the signals. Some of the commonly used time-frequency techniques are: short-time Fourier transform (STFT), and Wavelet transform (WT). STFT is the FT of the product of the signal and shifted version of the window function (rectangular, Hanning, and Hamming window). The STFT uses fixed window length. To overcome this issue, variable window length is needed and hence the wavelet transform has been introduced.

2.7.1 Short-Time Fourier Transform (STFT)

The short-time Fourier transform (STFT) is a Fourier-related transform that can be used to identify the sinusoidal frequency and phase content of local parts of a signal as they change over time. In reality, computing STFTs entails dividing a larger temporal signal into equal-length segments and computing the FT independently on each shorter segment.

Figure 32 Time-frequency plane of STFT

Adapted from:
https://www.mathworks.com/help/examples/wavelet/win64/TimeFrequencyAnalysisUsingContinuousWaveletTransformExample_03.png (**Accessed on; Jan 15th 2021**)

The signal $x(t)$ is multiplied by a window function that is nonzero for just a short duration. As the window $w(\tau)$ is moved along the time axis, the FT $X(\tau, \omega)$ of the resulting signal is computed, yielding a 2D representation of the signal. This is written mathematically as:

$$X(\tau, \omega) = \int_{-\infty}^{\infty} x(t)w(t-\tau)e^{-i\omega t}\, dt \qquad (3)$$

The STFT is difficult to apply to non-stationary signals; there is no way to resolve the complete frequency content of such signals with a single window size [**76**].

2.7.2 Continuous Wavelet Transform (CWT)

The continuous wavelet transform (CWT) was formalized by Morlet and Grossman in 1984 [**77**], to address the STFT's resolution difficulties. A wavelet is a tiny waveform that meets two criteria. The signal's amplitude should rapidly fall and converge to zero in the first case, and the function should fluctuate in the second. A feature of the wavelet filter is the ability to split a signal into many frequencies and remove a specific frequency and amplitude. To breakdown a signal into wavelets, the CWT is utilized. The phrase "continuous wavelet" refers to its ability to scale to any time scale [**69**]. The CWT on the time-frequency plane is shown in Fig. 33.

Figure 33 Time-frequency plane of CWT

Adapted from:
https://www.mathworks.com/help/examples/wavelet/win64/TimeFrequencyAnalysisUsingContinuousWaveletTransformExample_06.png (**Accessed on; Jan 15th 2021**)

The CWT's basis functions are scaled and shifted versions of the mother wavelet. The CWT provides excellent time and frequency localization [**78**]. The $CWT\{x(t), a, b\}$ is defined as the integration of the original sequence $x(t)$ with the shifted or scaled shape of the mother wavelet $\psi_{a,b}(t)$:

$$CWT\{x(t), a, b\} = \frac{1}{\sqrt{|a|}} \int_{-\infty}^{\infty} x(t)\psi(\frac{t-b}{a})dt \quad \text{where a, b} \in R, \text{a} \neq 0 \qquad (4)$$

Here a and b are the scale and position parameters respectively. The CWT produces a matrix containing wavelet coefficients located by scale and position. The mother wavelet has a significant role and there are different functions that can be used as a mother wavelet [**79**]. An analytic wavelet is a complex-valued wavelet where Fourier transform is only valid on the positive real axis, and Morse wavelet is one of this family. This is helpful to analyze signals that change in amplitude and frequency over time, as well as signals with localized discontinuities [**80**].

2.7.3 Stockwell Transform (S-transform)

The Stockwell (S) Transform is a hybrid combination of STFT and wavelet transform that takes the advantage of both [**18**] [**25**]. The S-transform is a time-frequency representation; it gives a full time-frequency or spatial-frequency decomposition of a signal. S-transform technique was proposed by Stockwell and his co-workers [**24**]. S-transform provides a good frequency resolution at low frequency and a good time resolution at a higher frequency, because of use of frequency-dependent window function.

The mathematical representation of the S-transform, $S(\tau, f)$: is expressed as:

$$S(b, f) = \frac{1}{\sqrt{|a|}} \int_{-\infty}^{+\infty} x(t)\psi(\frac{t-b}{a})\, e^{-j2\pi fb}\text{dt} \qquad (5)$$

where $x(t)$ is the input signal, f and b represent the frequency and the parameter which controls the position of the window on the y-axis, respectively. Deriving the S-transform as the continuous wavelet transform (CWT) phase correction is thought to be instructive. The CWT of a function $x(t)$ is defined by:

$$CWT\{x(t), a, b\} = \int_{-\infty}^{+\infty} x(t)\, \psi_{a,b}(t)\ dt \qquad (6)$$

where $\psi_{a,b}$ is a scaled replica of the mother wavelet. The S-transform of a function $x(t)$ is defined as a CWT with a specific mother wavelet multiplied by the phase factor.

$$S(b, f) = e^{-i2\pi fb} CWT\{x(t), a, b\} \qquad (7)$$

CHAPTER 3

Literature Review

3.1 Introduction

The abnormal conditions and malfunctions of the heart can be detected from ECG signals by recording the potential bio-electric variation of the heart. Automatic identification of ECG signals has been tried on several studies before using computer aided diagnosis systems [**20**]. In this regard, several researches have been conducted by using different machine learning techniques [**81**] [**82**]. Different methods for classification of ECG signals have been presented in previous studies making use of, for example, ANNs [**83**] and statistical methods [**84**].

One study reported an accuracy of 90.6% for ECG classification into six different classes based on ANN [**85**]. In another study, the detection of four arrhythmia types achieved an average accuracy of 96.95% by using a feed-forward neural network as a classifier [**86**]. Feature extraction using a recurrent neural network (RNN) was employed in another study [**87**] and resulted in 98.06% accuracy for four classes of arrhythmias. For classifying 1-D electrocardiogram signals, 1-D CNN models were proposed [**88**] [**89**] in two different studies and they reported accuracy of 96.72% and 97.03% respectively. In a separate study, a 2-D CNN model with nine layers was applied for classification of five different heartbeat arrhythmia types, and the accuracy was 94.03% [**90**].

3.2 Deep Learning

A deep neural network is a type of machine learning that employs multiple layers of computation to create a deep neural network; it is capable of learning from large amounts of data. Deep learning approaches have recently demonstrated their superior performance. As a result, scientists and engineers are focusing their efforts on ECG classification studies using deep learning algorithms. Many researchers have focused their efforts on applying deep learning approaches to examine ECG categorization. In the literature, several approaches for ECG arrhythmia categorization have been reported [**91**] [**92**] [**93**].

3.3 Transfer Learning

Different methodologies were investigated for the literature review in the neural networks arena in order to get towards classification processes. Transfer learning distinguishes out among the different current neural network algorithms in terms of time savings and

versatility. Transfer learning allows a model built on related data to be utilized for another feature space [**18**]. Training a deep CNN from scratch is time-consuming and needs significant amount of data. When there is no enough data available for training, it's preferable to use existing neural networks that have been trained on huge data sets for jobs that are conceptually comparable.

Transfer learning was used for the detection of arrhythmia, a pre-trained classifier AlexNet was employed to classify three different ECG beats: normal, right bundle branch blocks and paced. MIT-BIH database was used for the training and validation purposes. The study reported 92% accuracy rate [**94**].

3.4 Machine Learning Applications

Machine Learning is mainly used for classification, prediction and clinical diagnosis in areas of cardiovascular issues. Jinkwon Kim and Hang Sik Shin proposed an algorithm that is able to classify six different types of heart beats. They made a comparative study of algorithms using an Extreme Learning Machine (ELM), radial basis function network, back propagation neural network, and support vector machine (SVM) [**95**]. Their study concluded that ELM was able to perform a fast learning speed and offered high performance. They developed an algorithm by analyzing the proper signal processing methods for each part of arrhythmia classifier. Morphology Filter (MF) was chosen for the pre-processing step to remove noise and preserve the morphology of ECG signals. For the classification part, they reduced the learning time and maintained high accuracy by using ELM. In the evaluation stage, the entire MIT-BIH arrhythmia database was used to maintain the robustness of the algorithm. This algorithm showed effective accuracy and performance with a short learning time and all beat types were classified with high accuracy.

3.5 Artificial Neural Network Applications

Applications of ANN were mainly found to be classification, prediction, and diagnosis. Some of the applications in classification include: using ANN for organizing or distinguishing the data in medical databases by relevant categories or concepts [**96**], automatic tissue recognition in wound images using a hybrid learning approach for accurate wound evaluations [**97**], and comparison of computing techniques used in diagnosis of heart conditions (normal and abnormal) by processing recorded heart sound signals to extract the features (time and frequency) [**98**]. Some applications in prediction include developing a model to predict the chances of diabetes complications based on risk factors changes [**99**] and

in the emergency department modelling daily patient arrivals [**100**]. In diagnosis, ANN has been used in diagnosis of coronary artery diseases [**101**], and diagnosis for pulmonary tuberculosis [**102**].

The first CNN-based algorithm was introduced in 1989 [**103**]. This algorithm was proposed and used to recognise handwritten zip codes. Several CNN classifiers have been developed for images classification. The process used in the feed-forward classifiers does not consider the local spatial information. By developing CNN models, spatially adjacent pixels can be used for local features extraction from 2D images.

CNN can use short-term ECG signals for analysing their characteristics and it has been recently used for multi-dimensional inputs (1D, 2D and 3D) [**104**]. 1D CNNs are less versatile than 2D CNNs, and transforming the 1D data into a 2D format could be helpful in particular tasks [**105**]. Urtnasan Erdenebayar et al. [**106**] proposed a CNN model for automatic prediction of atrial fibrillation (AF) using short-term (30 seconds) ECG signals. Pre-processing and segmentation has been done before training the network. Two AF and one normal dataset from the MIT-BIH database were used. The method resulted in 98.6% sensitivity, 98.7% specificity, and 98.7% accuracy. For pre-processing, the elimination of the baseline and high-frequency noises from the input signals was done by using band-pass filter and discrete wavelet transform. The network consists of three main layers: convolutional layer, max-pooling layer, and fully-connected layer. The fully-connected layer has a fully-connected multi-layer perceptron (MLP) and soft-max activation function. The researchers selected seven-layer network as the optimal architecture of this model. The model could perform robustly for the automatic prediction of AF (see also Fig. 34).

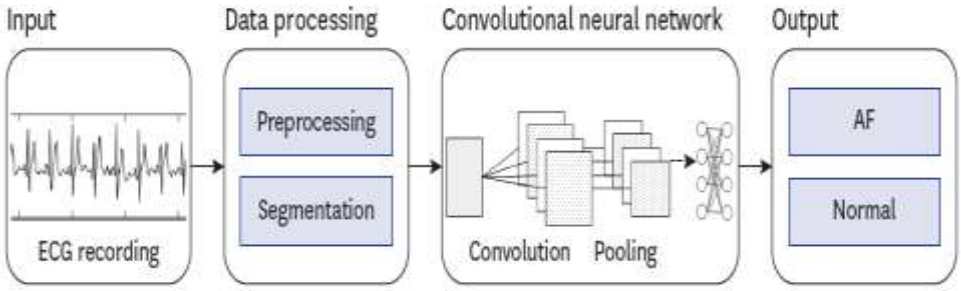


Figure 34 Diagram of proposed method for automatic prediction of Atrial Fibrillation

Adapted from:https://doi.org/10.3346/jkms.2019.34.e64(**Accessed on; Nov 3rd 2020**)

Amin Ullah et al. [**21**] proposed a 2D CNN as a classifier for eight ECG signal types: normal beat, premature ventricular contraction beat, paced beat, right bundle branch block beat, left

bundle branch block beat, atrial premature contraction beat, ventricular flutter wave beat, and ventricular escape beat. As also been shown in Fig. 35, the 1D ECG signals were first transformed into 2D images using short-time Fourier transform. The classifier has four convolutional layers and four pooling layers. The online MIT-BIH arrhythmia dataset were used and 97.91% sensitivity, 99.61% specificity and classification accuracy of 99.11% were achieved in this study.

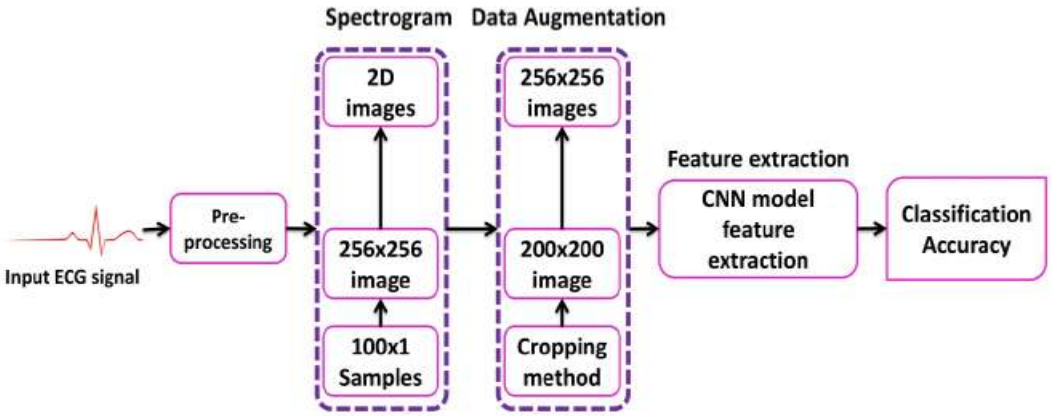


Figure 35 Complete procedure of ECG signal classification proposed in arrhythmias classification study

Adapted from:doi:10.3390/rs12101685(**Accessed on; Nov 3rd 2020**)

Awni Hannun [**19**] developed a 34-layers deep neural network that was able to classify twelve different arrhythmias. Single lead ECG signals collected from 53,549 patients were used in their study. The network was validated using independent dataset consisted of ECG signals collected from 328 patients, and resulted in 83.7 F1-score which exceeded the average cardiologist F1-score (78). The study demonstrated that deep learning can be used to classify different types of arrhythmia and achieve performance that is similar to the cardiologist's performance, and if it is applied in clinical settings, misdiagnose can be reduced. They recommended having additional training before using this approach in clinical applications.

There are also other studies that used deep learning algorithms based on CNNs to classify cardiac arrhythmias based on ECG signals. Table 1 compares some of the models utilized, and the results produced.

Table 1 Basic techniques used in classifying ECG signals using CNN

Researcher	Preprocessing	Dataset	# of Classes	Model	Accuracy
Roberta et al. [**17**]		MIT-BIH arrhythmia	3	5-layers CNN	98.33%
Dan Li et al. [**107**]	Wavelet transform	MIT-BIH arrhythmia	5	7-layers CNN	97.50%
U. Rajendra Acharya et al. [**108**]	Augmentation and Digital Filtration	MIT-BIH arrhythmia	5	9-layers CNN	94.03%
Amin Ullah et al. [**21**]	STFT	MIT-BIH arrhythmia	8	10-layers CNN	99.11%
Özal Yıldırım et al. [**23**]	Normalization and Rescaling	MIT-BIH arrhythmia	13 15 17	16-layers CNN	95.20% 92.51% 91.33%

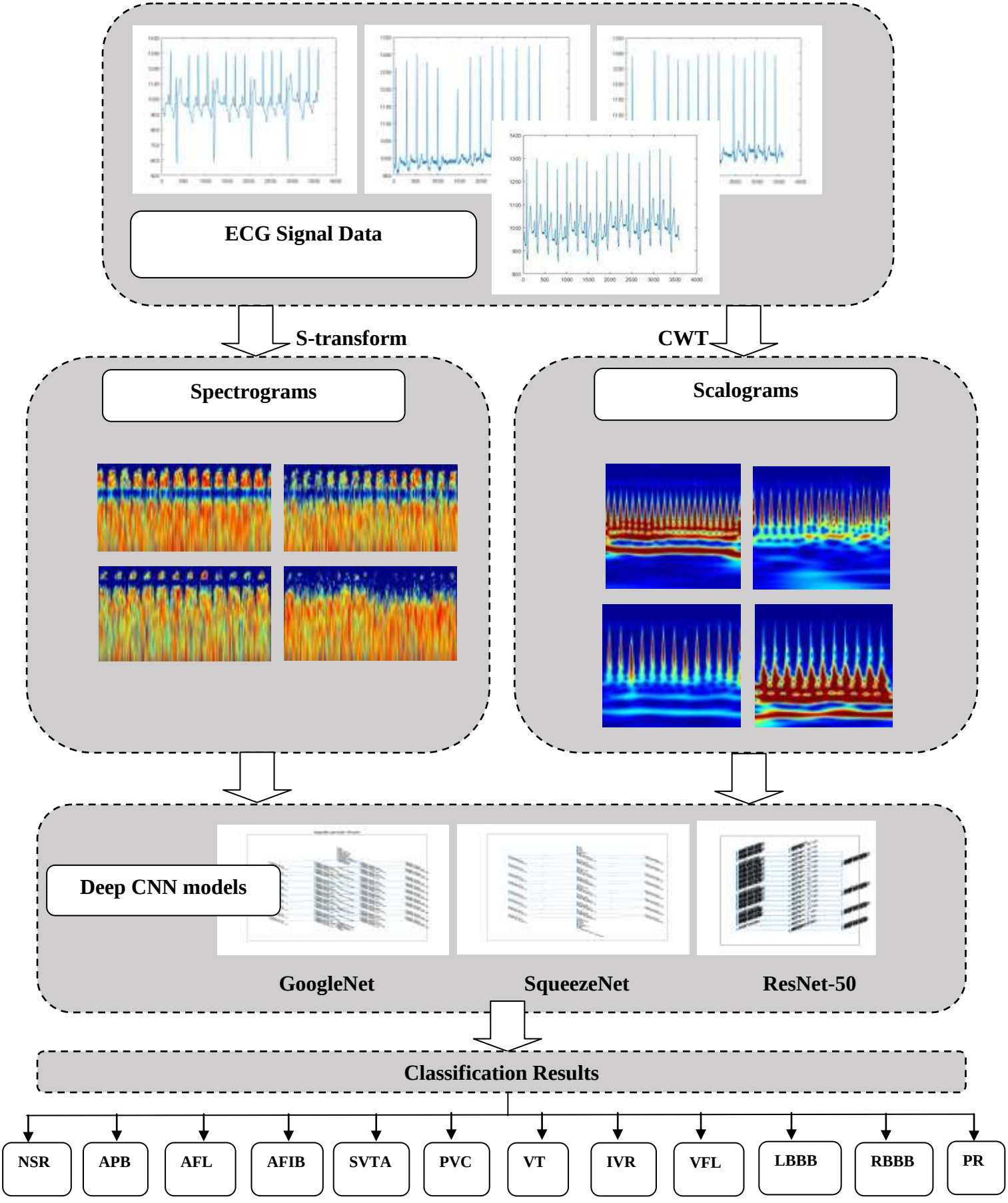
CHAPTER 4

Material and Methods

This chapter provides information about the study design and population, data pre-processing, processing and classification of the different types of arrhythmias using several CNN architectures. The proposed methodology primarily targets to compare the performance of different pretrained CNNs on classifying eleven arrhythmia classes and the ground truth based on the CWT and the S-Transform. The different arrhythmia classes are: Atrial Premature Beat (APB), Atrial Flutter (AFL), Atrial Fibrillation (AFIB), Supraventricular Tachyarrhythmia (SVTA), Premature Ventricular Contraction (PVC), Ventricular Tachycardia (VT), Idio-ventricular Rhythm (IVR), Ventricular Flutter (VFL), Left Bundle Branch Block Beat (LBBBB), Right Bundle Branch Block Beat (RBBB), and Pacemaker Rhythm (PR). A schematic diagram for the flow of the proposed method is depicted below.

Experimental studies in this research were performed on a computer with Processor Intel(R) Core(TM) i7-4600U CPU @ 2.10GHz 2.70 GHz, Installed RAM 12.0 GB, and System type 64-bit operating system, x64-based processor, and the models were implemented and trained using MATLAB R2020a. Several quantitative metrics were used to evaluate the model's performance: accuracy, precision, sensitivity, specificity, F1-score, and classification time. The classification using several pre-trained networks: GoogleNet, SqueezeNt, and ResNet-50 were compared. Impacts of using the S-transform and CWT in generating the 2D representation from the 1D ECG signals are presented in this research study. The study compares the affect of using images generated from S-transform and CWT in different classifier's performance.

The performance of the proposed algorithm in accurately classifying eleven arrhythmia classes was verified against the ground truth information (normal sinus rhythm records) which is also available in the same dataset. Prior to classification, the 1D signals were processed using the CWT and the S-transform, then the transformed images were resized to size 224×224 or 227×227 to be appropriate for the CNN networks utilized in the study.


ECG Signal Data
S-transform
CWT
Spectrograms
Scalograms
Deep CNN models
GoogleNet
SqueezeNet
ResNet-50
Classification Results
NSR
APB
AFL
AFIB
SVTA
PVC
VT
IVR
VFL
LBBB
RBBB
PR

4.1 Data Set

The cardiac arrhythmia data set used for the classification of the ECG fragments in this research study was extracted from a subset of the open-source MIT/BIH arrhythmia data set. Prior to the availability of this data for research purpose, the records had undergone basic pre-processing such as band pass filtering of 0.1 to 100 Hz frequencies [**109**]. This subset contains a total of ten seconds 1000 randomly-selected signals collected from 45 patients, including 17 classes of cardiac arrhythmia. Twelve classes were chosen from this subset with a total number of 890 signals (i.e. Normal Sinus Rhythm and 11 types of heart arrhythmias). The sampling frequency was 360 Hz using single lead position (MLII) among many patients. This subset of the MIT-BIH data set is freely available for use as an online resource at: https://data.mendeley.com/datasets/7dybx7wyfn/3. Table 2 displays the distribution of cardiac arrhythmia classes found in the 1000-ECG signals.

Table 2 Distribution of ECG fragments used for the various classes

No	Class	# of Signals	Classes	
			Training	Validation
1	Normal sinus rhythm (NSR)	283	226	57
2	Atrial premature beat (APB)	66	53	13
3	Atrial flutter (AFL)	20	16	4
4	Atrial fibrillation (AFIB)	135	108	27
5	Supraventricular tachyarrhythmia (SVTA)	13	10	3
6	Pre-excitation (WPW)	21		
7	Premature ventricular contraction (PVC)	133	106	27
8	Ventricular bigeminy	55		
9	Ventricular trigeminy	13		
10	Ventricular tachycardia (VT)	10	8	2
11	Idioventricular rhythm (IVR)	10	8	2
12	Ventricular flutter (VFL)	10	8	2
13	Fusion of ventricular	11		
14	Left bundle branch block beat (LBBBB)	103	82	21
15	Right bundle branch block beat (RBBBB)	62	50	12
16	Second-degree heart block (SDHB)	10		
17	Pacemaker rhythm (PR)	45	36	9
	Total	1000	711	179

4.2 Pre-processing

The ECG signal illustrates non stationary data where the frequency characteristics changes with time. Using frequency domain information cannot fully represent the changes. The 1D signal can be transformed from the time domain to the joint time-frequency domain to generate a 2D matrix using multi-resolution analysis. This was done in the current study using the CWT and the S-transform.

The CWT was used to create joint time frequency representation of the signal, which provides excellent time and frequency localization [**78**]. The CWT, $CWT\{x(t), a, b\ \}$, is defined as the integration of the original signal $x(t)$ with the shifted or scaled shape of the mother wavelet $\psi_{a,b}(t)$ as shown in Eq. 8:

$$CWT\{x(t), a, b\} = \frac{1}{\sqrt{|a|}} \int_{-\infty}^{\infty} x(t)\psi(\frac{t-b}{a})dt \quad , \text{a ,b} \in R \text{ , a} \neq 0 \qquad (8)$$

Here a and b are the scale and position parameters respectively. The CWT produces a matrix containing wavelet coefficients located by scale and position. There are different mother wavelets that could be used to compute the CWT of a given signal. In the current study, the CWT coefficients were computed using Morse wavelet. Many studies on Morse wavelet theory and its applications to signal analysis have been conducted [**110**] [**111**], as well in the area of ECG signal analysis [**112**]. One could get analytic wavelets that have different properties and behaviour by adjusting the time-bandwidth product and symmetry parameters of a Morse wavelet. In practice, a fine discretization of the CWT is computed where usually the position is discretized at the sampling interval and the scale is discretiszd logarithmically. Compared to the discrete wavelet transform, the CWT discretizes scale more finely [**113**].

The S-transform takes the advantage of STFT and wavelet [**24**] [**25**]. S-transform provides a good frequency resolution at low frequencies and a good time resolution at higher frequencies, because of use of frequency-dependent window function. The one dimensional ECG signals $x(t)$ are converted into two dimensional spectrograms by applying the S transform $S(b, f)$ as follows:

$$S(b, f) = \frac{1}{\sqrt{|a|}} \int_{-\infty}^{+\infty} x(t)\psi(\frac{t-b}{a})\, e^{-j2\pi fb}\mathrm{dt} \qquad (9)$$

where f and b represent the frequency and the parameter which controls the position of the Gaussian window (width factor 1) on the y-axis, respectively.

Using Image Batch Processor application in MATLAB R2020a and setting resize function, all scalograms and spectrograms were resized into 224×224 and 227×227 to be appropriate for the use by the proposed CNN networks.

4.3 Convolutional Neural Networks and Classification

After the signals are transformed into images, they are inputted into classifiers to categorize them into normal class or any of the other eleven arrhythmia classes. Several methods have been used in the literature for detection and classification of arrhythmia, though the most commonly used classifier is CNN.

To execute classification on data sets (binary vs higher category sets) in various combinations, different primary classification groups were constructed. As a result, network designs were modified in accordance with the dimensions of the incoming data set. The models were implemented using MATLAB R2020a. Three deep CNNs were used: GoogleNet, SqueezeNet and ResNet-50 and all classifiers were trained to identify ECG waveforms based on a time-frequency representation. The used deep CNNs were initially created to classify images into 1000 classes. We employed the CNN's network architectures to identify ECG signals based on images generated from the CWT and the S-transform of the time series data. Splitting the data into training and testing is important before starting the classification. In this regard, the images were randomly divided into two groups: 80% of the images used for training, and the remaining 20% for validation. The optimizer that was used for training was stochastic gradient descent with momentum (SGDM).

4.3.1 CNN Models

Images were inputted into classifiers to categorize them. To execute the binary and multi-class categorization of the data set, the data were combined into three groups. For binary classification, the data were labelled into two groups: Normal and Arrhythmia. The second group had the signals from the NSR and eleven arrhythmia types: APB, AFL, AFIB, SVTA, PVC, VT, IVR, VFL, LBBBB, RBBB, and PR. For the purpose of comparison with pervious works [**21**], last group were created containing eight classes of arrhythmia: NSR, APB, IVR, LBBB, PR, PVC, RBBB, and VFL. Classifying the arrhythmia into different classes was done using three different pre-trained networks: GoogleNet, SqueezeNet and ResNet-50.

4.3.1.1 GoogleNet

GoogleNet architecture is based on Inception design [**114**], which enables the network to employ many sorts of filter sizes in a single image block (filters can function on the same

level), rather than being limited to a single filter size, which will then concatenate and pass onto the next layer [**115**]. Instead of being deeper, the network expands wider with this concept (see also Fig. 36).

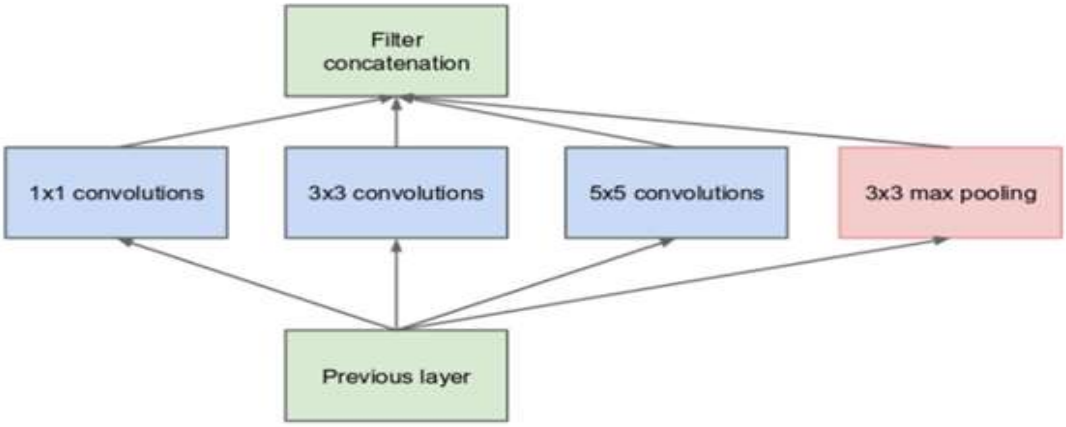


Figure 36 Inception module for GoogleNet, where the convolution operation performed using different filter sizes

Adapted from: https://images.deepai.org/django-summernote/2019-06-18/5ebad056-29d3-4f4c-bef1-2f262388afb0.png (**Accessed on; Oct 6th 2021**)

GoogleNet is a deep CNN with 22 layers. The network's convolutional layers extract features, which are then used by the final classification layer to categorize the image (see Fig. 37).

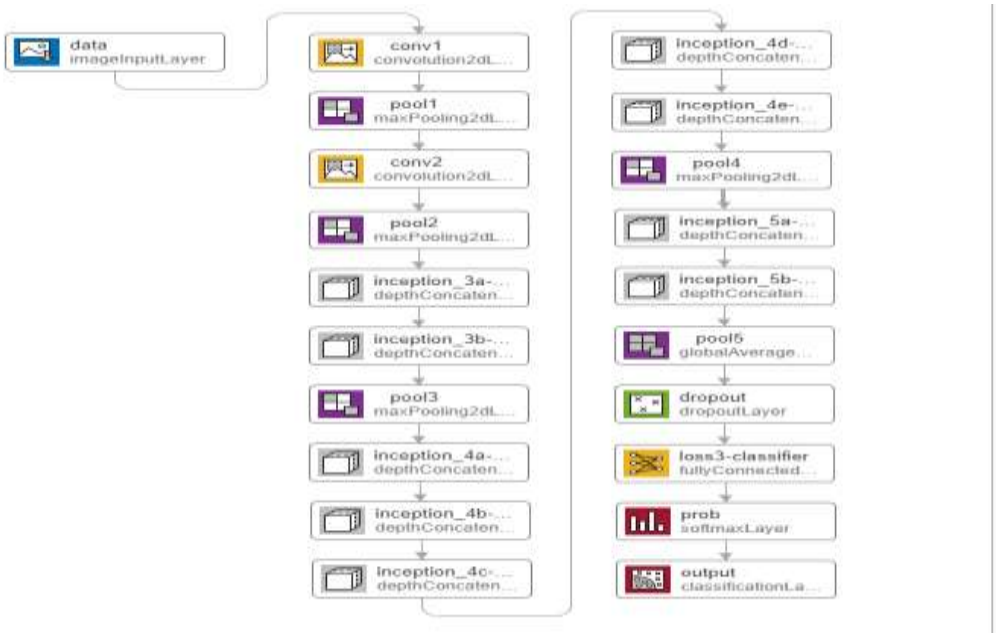


Figure 37 GoogleNet architecture (generated in MATLAB)

A deep neural network GoogleNet was used for the classification of image data set in the current study. The input size of 224×224×3 was selected. For implementing GoogleNet in this study, the final dropout layer probability was set to 0.6, in order to prevent over fitting. The classification layer and the last fully connected layer were replaced because they contain information about the data that initially classified such as the predicted labels. The last fully connected layer was replaced with a new one and has number of neurons equal to the number of the new classified categories.

4.3.1.2 SqueezeNet

In 2016, researchers from DeepScale, the University of California, Berkeley, and Stanford University presented architecture of SqueezeNet, which is a smaller network that was created to be a more compact alternative to AlexNet. It has less parameters than AlexNet yet performs faster [**116**]. The "squeeze" and "expand" layers make up the fire module which is the base of the SqueezeNet architecture. A squeeze layer has only 1 × 1 filters, which will then fed into an expand layer that has a mix of 1 × 1 and 3 × 3 convolution filters as also been depicted in Fig. 38.

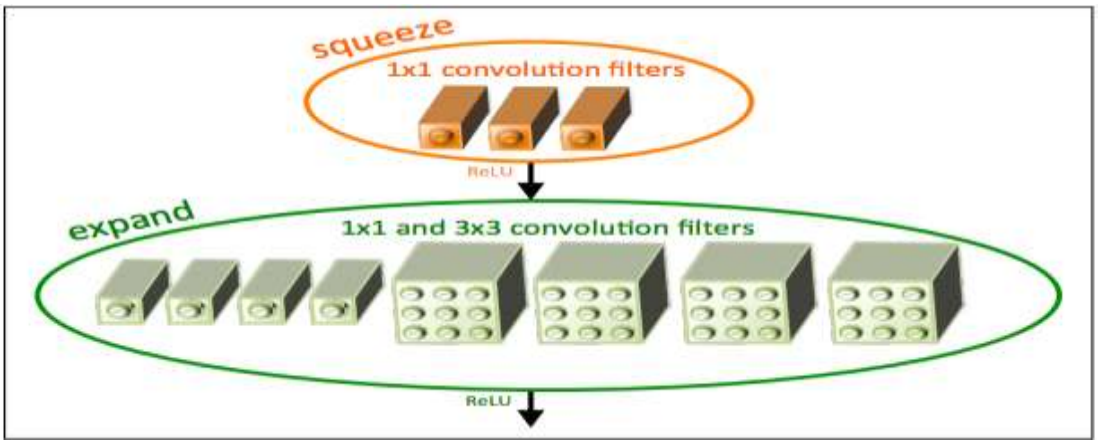


Figure 38 The fire module used in SqueezeNet

Adapted from: https://www.researchgate.net/profile/Pornntiwa- https://cdn-images-1.medium.com/max/1000/1*C4Y78hoaN0hPxyWJnkG5vQ.png (**Accessed on; Oct 6th 2021**)

A convolutional layer is used to process an input initially, followed by eight "fire modules" labeled "fire2-9". After that, "simple bypass" is used to increase the number of filters per fire module. Finally, following these layers, SqueezeNet executes max-pooling with a stride of 2. The architecture of the SqueezeNet network has been depicted in Fig. 39.

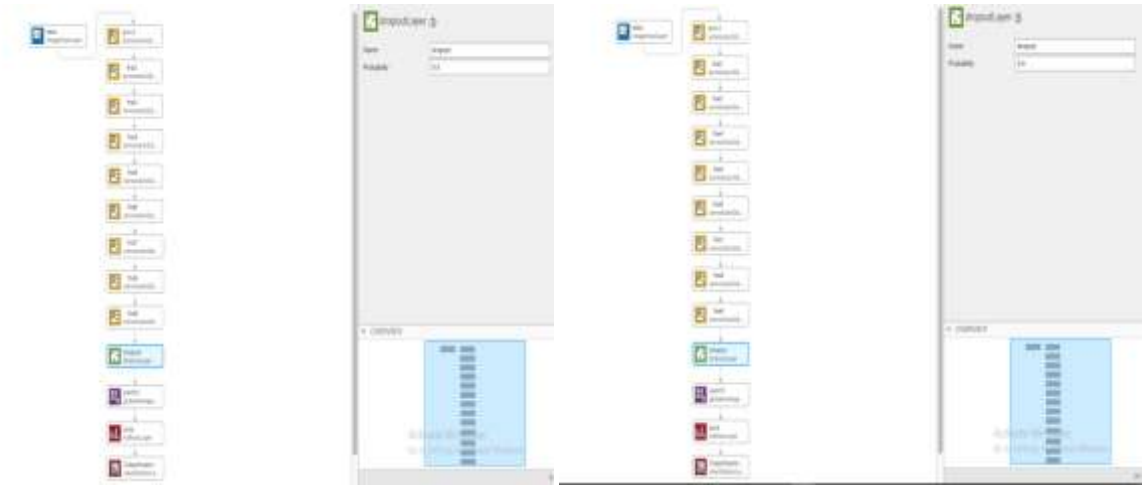


Figure 39 SqueezeNet architecture (generated in MATLAB): a) with dropout layer 0.5, b) dropout 0.6

Within the fire module, the ReLU activation is applied to the entire squeeze and expand layers. After the fire9 module, dropout layers are added to reduce overfitting, with a probability of 0.5. In the network, there are no dense layers [**116**].

SqueezeNet is a deep CNN with an architecture that can handle images with dimension of 227 x 227 x 3. The first step was to resize the spectrograms and scalograms into the expected input size by using augmented image data store. Some modifications were done to retrain SqueezeNet; the probability of the last dropout layer was set to 0.6. SqueezeNet's final learnable layer is a 1-by-1 convolutional layer, rather than a fully connected layer. Substituting new convolutional layer with the same number of filters as the number of classes was used to replace 'conv' layer. The classification layer was changed into a new one without class labels.

4.3.1.3 ResNet-50

Even though deep learning models produce promising results, there is a problem with more layers causing greater loss, which means that deeper networks will eventually have lower training accuracy. Deep residual learning, fortunately, solves the above-mentioned complexity concerns. A network output in residual learning learns not only from the activation of the previous layer, but also from the original input to that layer. ResNet-18, ResNet-34, and ResNet-50 are some examples of frequent residual deep learning networks [**117**]. Figure 40 shows the basic principle of ResNet.

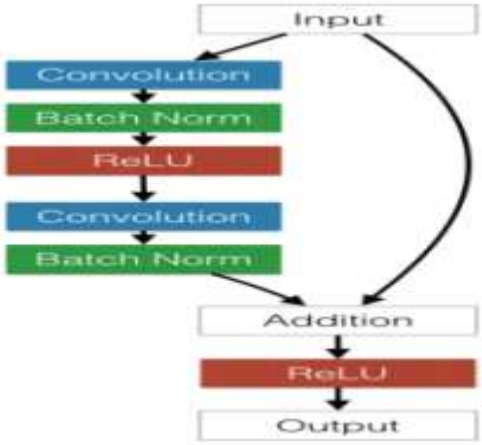


Figure 40 ResNet basic block

Adapted from: https://miro.medium.com/max/375/1*5UAu2uxrMIi4EAaII6IY8g.png (**Accessed on; Oct 4th 2021**)

ResNet-50 is a fifty-layer deep CNN that has been pre-trained to categorize images into 1000 different groups. Layers included in the ResNet-50 model are batch normalization, 2D convolution, and max pooling. The network's image input size is 224 × 224 pixels. The scalograms and spectrograms from the main image data store were used which have the expected input size. For the implementation of modified ResNet-50 in the current thesis, a new dropout layer of probability 0.6 was added into the network before the last fully connected layer to prevent over-fitting. Last dense layers were removed and replaced with new, fully connected layer to classify signals into the intended number of classes, and classification layer was replaced with a new one without class labels (see also Fig. 41).

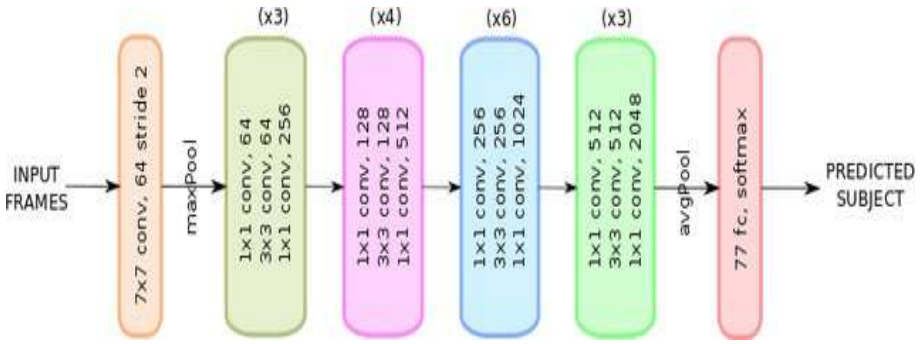


Figure 41 ResNet-50 architecture

Adapted from:
https://www.researchgate.net/publication/336805103/figure/fig4/AS:817882309079050@1572009746601/ResNet-50-neural-network-architecture-56_W640.jpg (**Accessed on; Oct 4th 2021**)

4.3.2 Evaluation Metrics

The classification performance of all the models was evaluated using the following metrics: accuracy, precision, sensitivity, specificity, F1-score, and classification time. Accuracy is the

ratio between the number of samples that are correctly classified and the total number of samples. The harmonic mean of the positive predictive value and sensitivity is known as F1-score.

$$\text{Accuracy} = \frac{TP+TN}{TP+TN+FP+FN} \quad (10)$$

where TP is the correct classification of arrhythmia, TN is the correct classification of normal beat, FP means the wrong classification as arrhythmia, and FN is the incorrect classification as normal [**118**].

$$\text{Precision (\%)} = \frac{TP}{TP+FP} \quad (11)$$

$$\text{Sensitivity (\%)} = \frac{TP}{TP+FN} \quad (12)$$

$$\text{Specificity (\%)} = \frac{TN}{TN+FP} \quad (13)$$

$$\text{F1-score} = 2 \times \frac{\text{Precision} \times \text{Sensitivity}}{\text{Precision} + \text{Sensitivity}} \quad (14)$$

CHAPTER 5

Results and Discussion

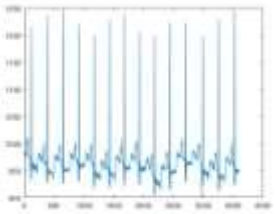
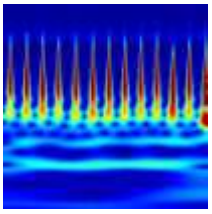
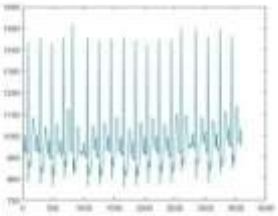
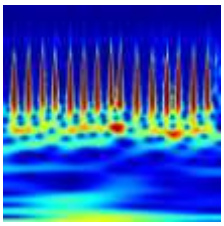
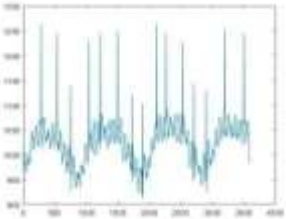
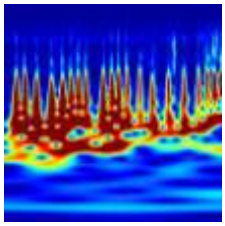
The experimental results presented below were acquired for both binary and multiclass classification using the proposed algorithm.

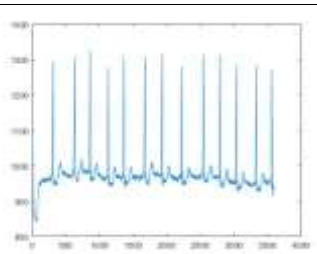
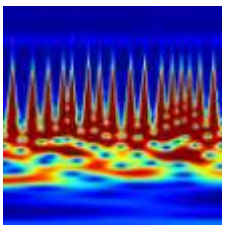
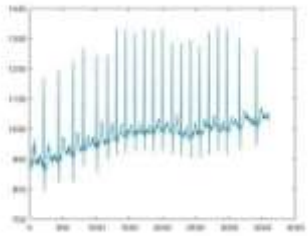
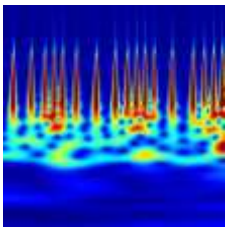
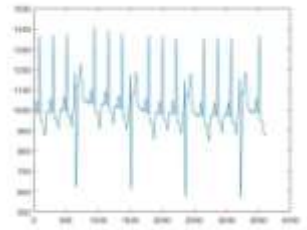
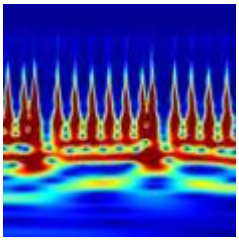
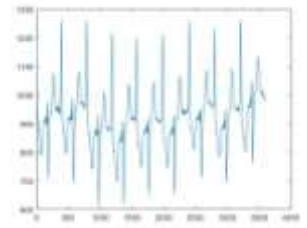
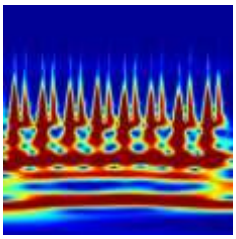
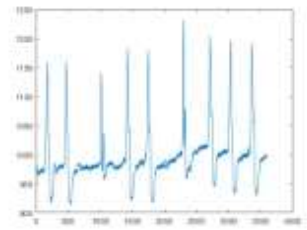
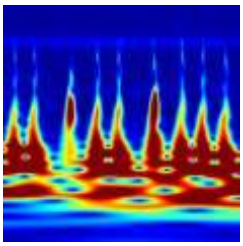
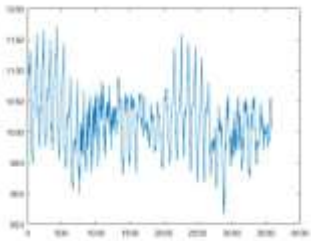
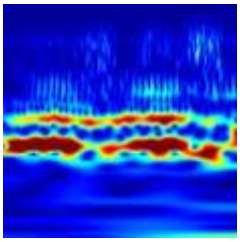
5.1 Pre-processing Results

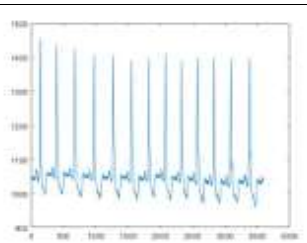
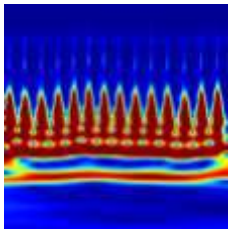
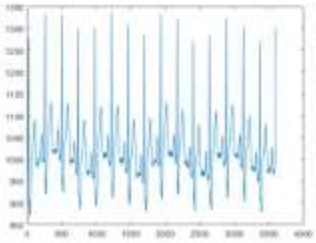
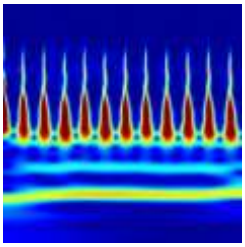
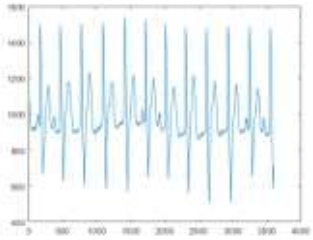
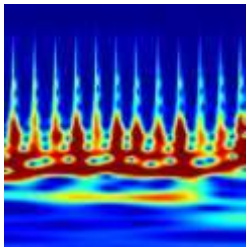
Time series signals were converted into time-frequency representation using CWT and the S-transform as presented in the Table 3 and Table 4.

Results Using Continuous Wavelet Transform (CWT):

Table 3 The Scalogram of each type of signals

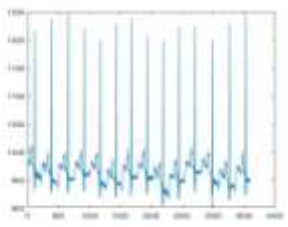
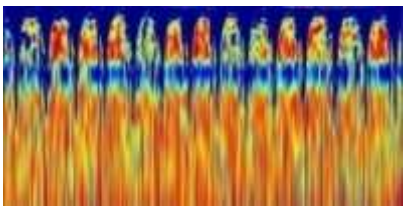
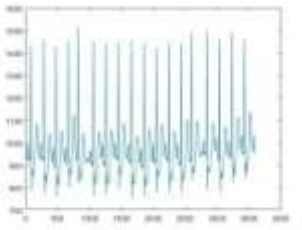
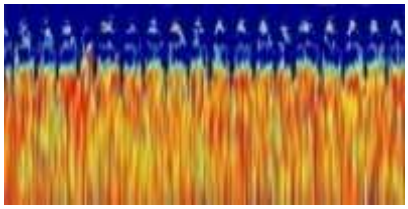
Type of Arrhythmia	Signal	Scalogram
NSR		
APB		
AFL		

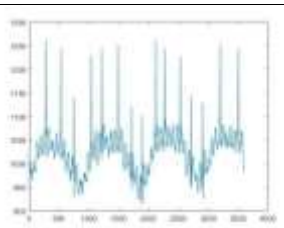
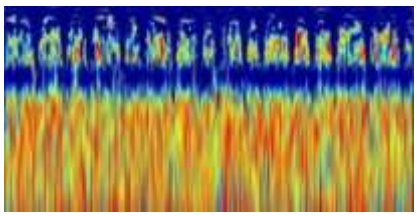
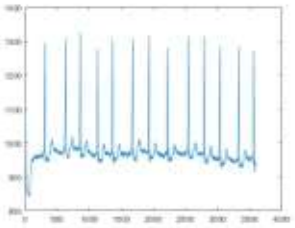
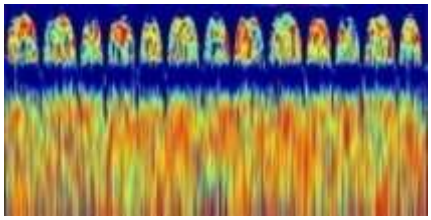
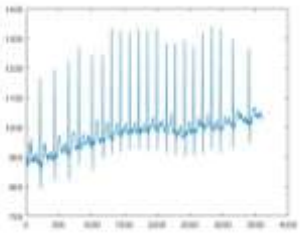
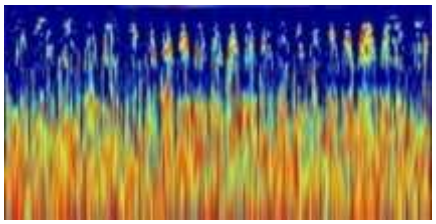
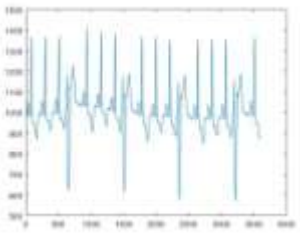
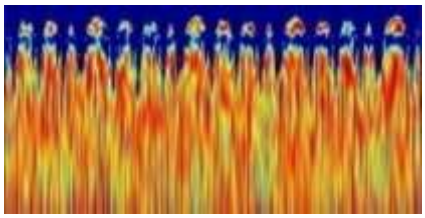
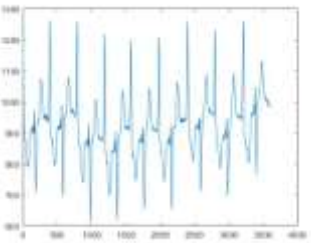
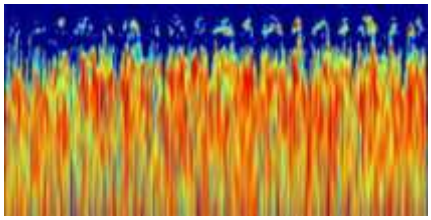
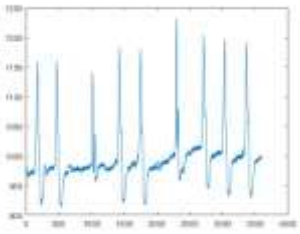
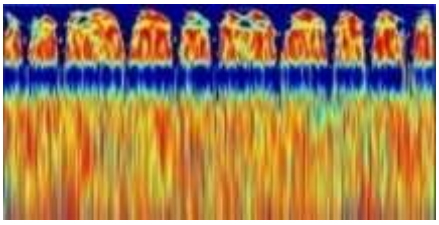
AFIB		
SVTA		
PVC		
VT		
IVR		
VFL		

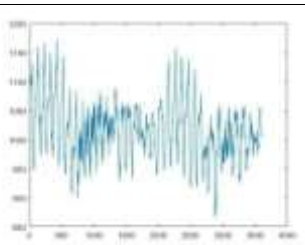
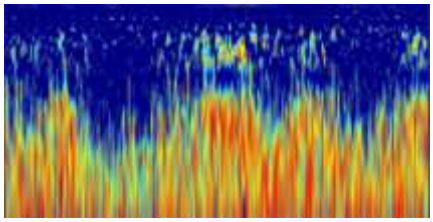
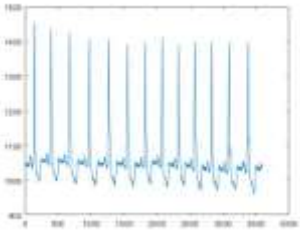
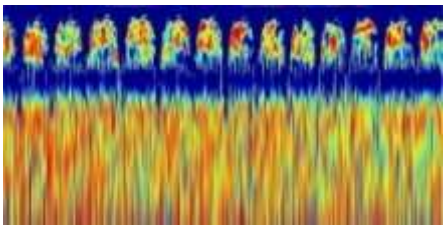
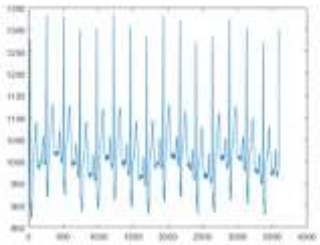
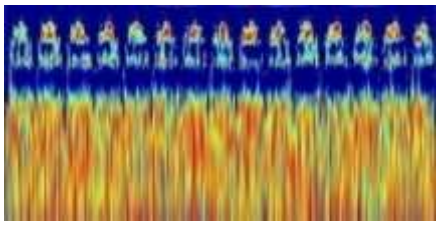
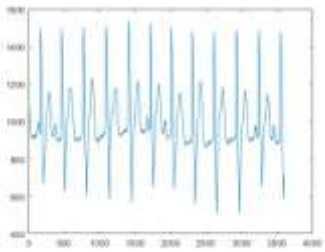
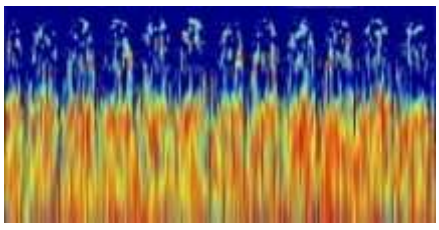
LBBB		
RBBB		
PR		

Results Using the S-transform:

Table 4 The Spectrogram of each type of signals

Type of Arrhythmia	Signal	Spectrogram
NSR		
APB		

AFL		
AFIB		
SVTA		
PVC		
VT		
IVR		

VFL		
LBBB		
RBBB		
PR		

5.2 Classification Results

A. Classification into Two Classes: Normal Rhythm and Arrhythmia

The used dataset from MIT-BIH contains observations that can be divided into two major groups: Normal and Arrhythmia. The binary classification was applied for these classes, as shown in Figure 44. In order to evaluate the performance of the binary classification, the confusion matrix was generated that allows compute the sensitivity, specificity, overall accuracy and other performance matrices. Figure 42 presents the confusion matrix generated as a result of using the proposed binary classification scheme developed in this thesis work.

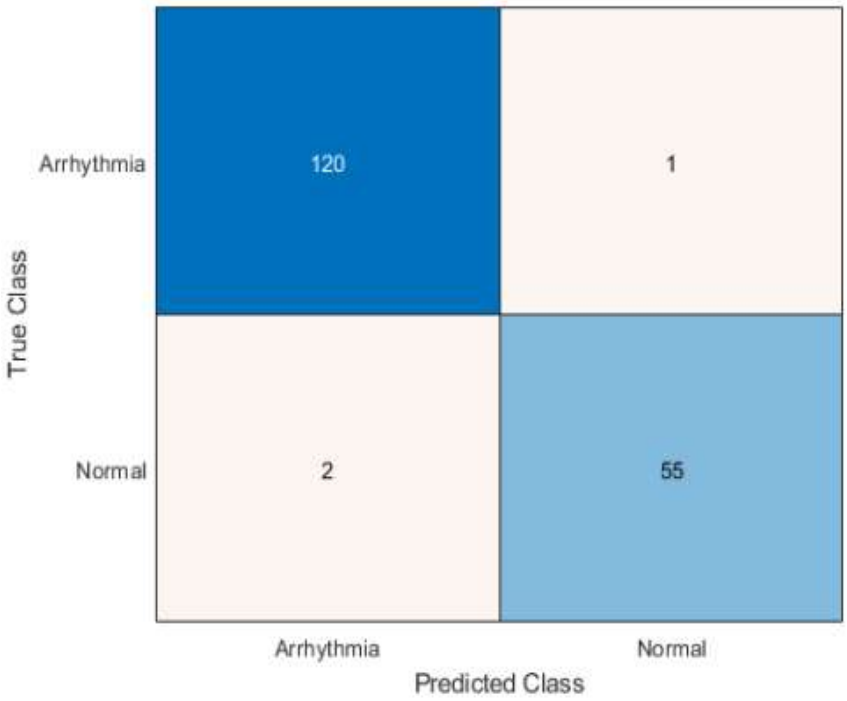


Figure 42 Confusion matrix for binary classification

From the generated confusion matrix, the accuracy, precision, sensitivity, f1-score, and specificity were calculated and those values are presented in Table 5. Accordingly the proposed binary classification scheme scored 98.3% precision, 97.9% sensitivity, 97.9% specificity, F1-score of 98.1 and 98.3% overall accuracy.

Table 5 Quantitative metrics for binary classification

Precision (%)	Sensitivity (%)	Specificity (%)	F1-score	Accuracy (%)	Time per Sample (ms)
98.3	97.9	97.9	98.1	98.3	0.154

B. Classification into 12 Classes

- **Results Using GoogleNet and the S-Transform**

For multiclass classification, GoogleNet was used to categorize images generated by using the S-transform into twelve including ground truth records. The progress of training the network has been depicted in Fig. 43. The validation data was used to evaluate the model's performance, and confusion matrix was generated. The confusion matrix is the one shown in Fig. 44. Accordingly, the scheme resulted in 81.5% precision, 72.1% sensitivity, 98.0% specificity, F1-score of 76.5 and 82.7% overall accuracy as also presented in Table 6.

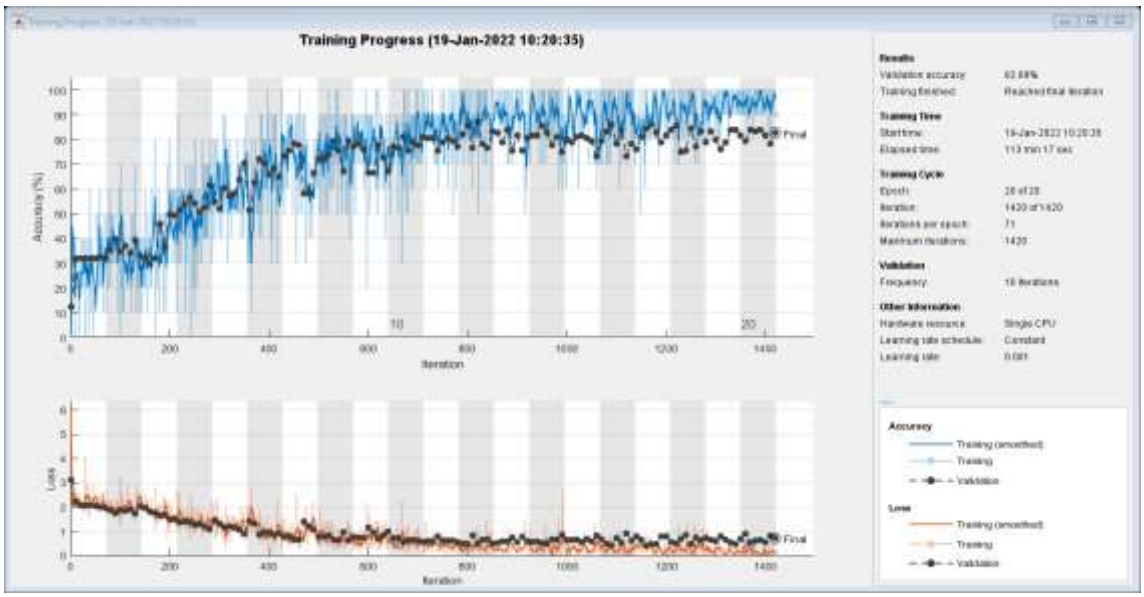


Figure 43 Accuracy and Loss values for classifying 12 arrhythmia using GoogleNet

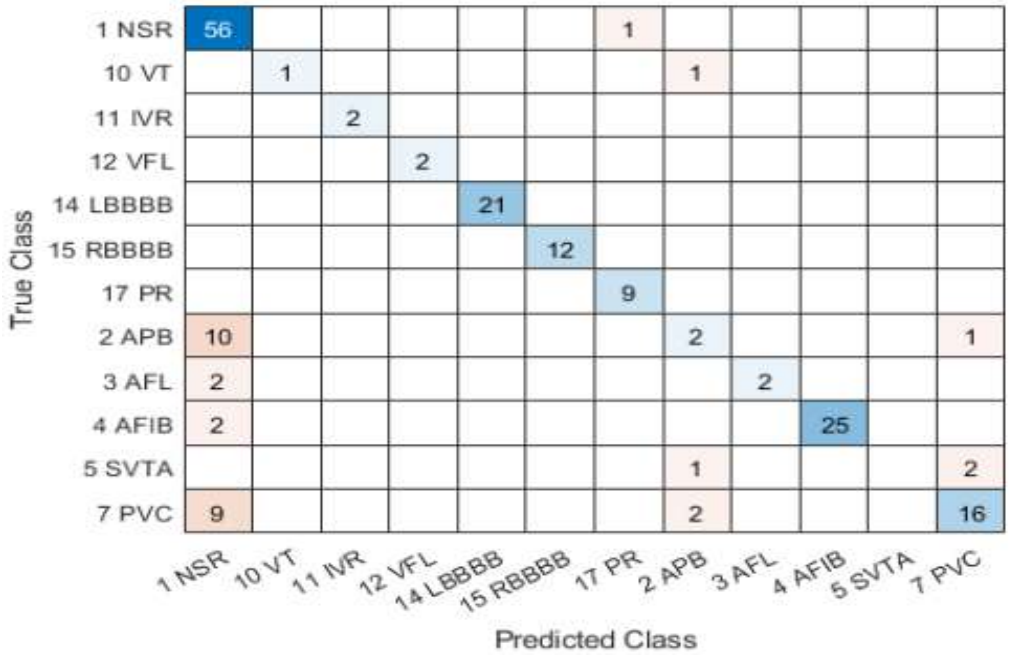


Figure 44 Confusion matrix for classifying 12 arrhythmia using GoogleNet

Table 6 Precision, sensitivity, specificity, F1-score , accuracy, and classification time of classifying 12 classes using GoogleNet and S-transform

Precision (%)	Sensitivity (%)	Specificity (%)	F1-score	Accuracy (%)	Time per Sample (ms)
81.5	72.1	98.0	76.5	82.7	0.170

- **Results Using SqueezeNet and the S-Transform**

The spectrograms generated by applying the S-transform were classified using SqueezeNet, and the training progress was recorded as displayed in Fig. 45. The performance of multiclass classification of spectrograms using SqueezeNet was evaluated using the validation data, and confusion matrix was generated. The confusion matrix is the one shown in Fig. 46. Accordingly, the scheme resulted in 83.3% precision, 70.4% sensitivity, 98.2% specificity, F1-score of 76.3, overall accuracy of 83.2% and classification time of 0.154 ms as also presented in Table 7.

Figure 45 Accuracy and Loss value for classifying 12 arrhythmia using SqueezeNet

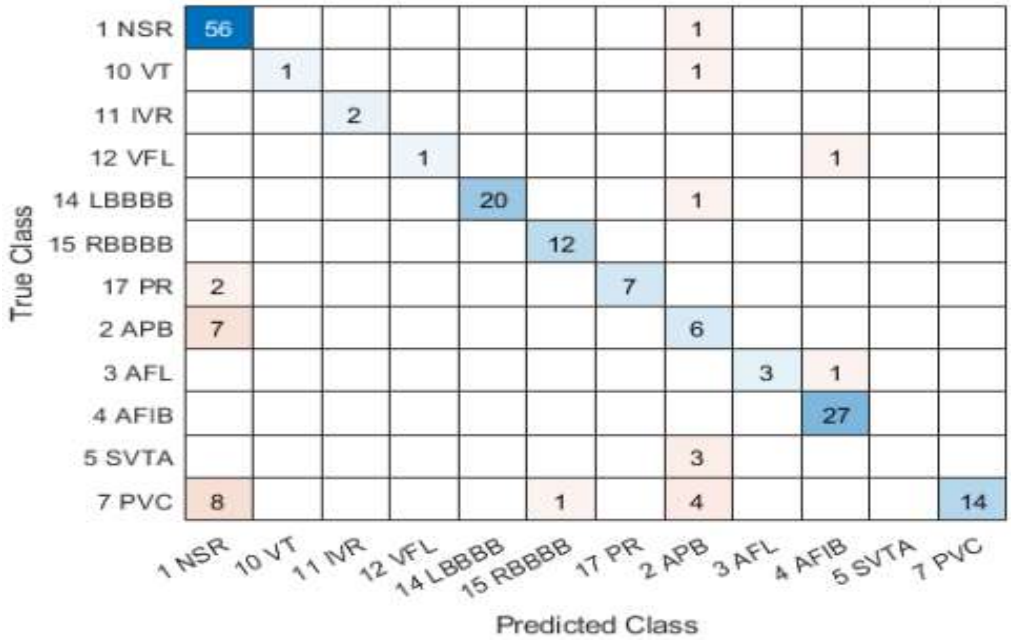


Figure 46 Confusion matrix for classifying 12 arrhythmia using SqueezeNet

Table 7 Precision, sensitivity, specificity, f1-score, accuracy, and classification time of classifying 12 classes using SqueezeNet and S-transform

Precision (%)	Sensitivity (%)	Specificity (%)	F1-score	Accuracy (%)	Time per Sample (ms)
83.3	70.4	98.2	76.3	83.2	0.154

- **Results Using ResNet-50 and the S-Transform**

ResNet-50 was used for multiclass classification to categorize spectrograms generated by using the S-transform into twelve including ground truth records. The progress of training the network has been displayed in Fig. 47. Figure 48 presents the confusion matrix generated for the ResNet-50 network for classifying 12 arrhythmia using validation data. Accordingly, the scheme resulted in 85.2% precision, 77.7% sensitivity, 98.8% specificity, F1-score of 81.3 and overall accuracy of 88.3% as also presented in Table 8.

Figure 47 Accuracy and Loss values for classifying 12 arrhythmia using ResNet-50

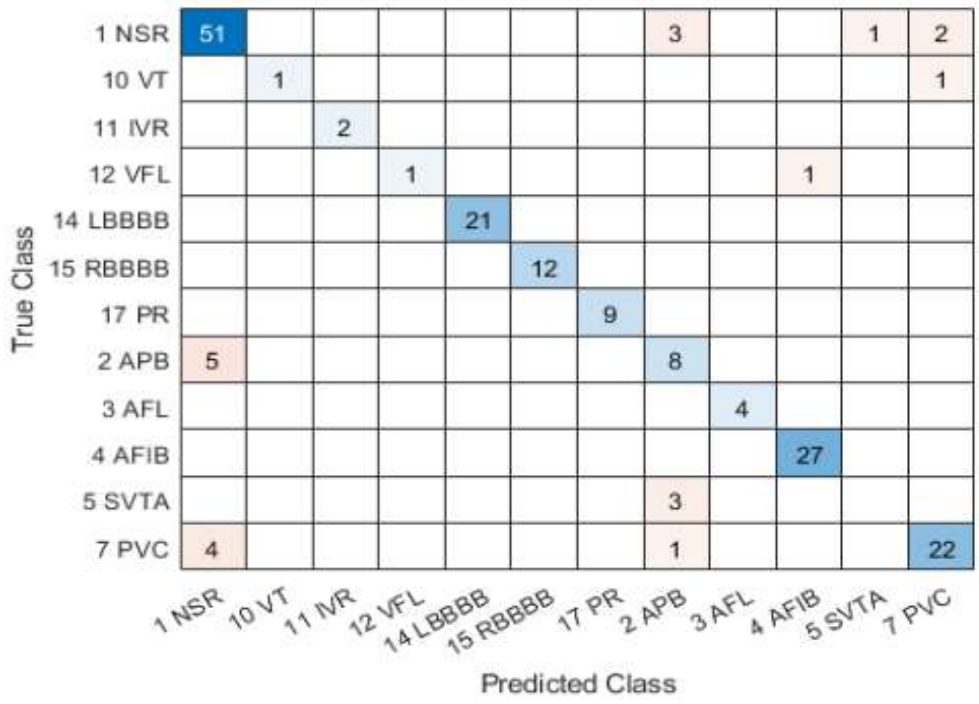


Figure 48 Confusion matrix for classifying 12 arrhythmia using ResNet-50

Table 8 Precision, sensitivity, specificity, f1-score, accuracy, and time of classifying 12 classes using ResNet-50 and S-transform

Precision (%)	Sensitivity (%)	Specificity (%)	F1-score	Accuracy (%)	Time per Sample (ms)
85.2	77.7	98.8	81.3	88.3	0.169

- **Results Using GoogleNet and the CWT**

Scalograms that resulted from transforming the time-series ECG data were inputted into GoogleNet for multiclass classification into twelve including ground truth records. The progress of training the network has been displayed in Fig. 49. Figure 50 presents the confusion matrix generated for the GoogleNet network for classifying 12 arrhythmia using validation data. Accordingly, the scheme resulted in 96.4% precision, 84.1% sensitivity, 99.4% specificity, F1-score of 89.9 and overall accuracy of 93.9% with 0.179 ms classification time as also presented in Table 9.

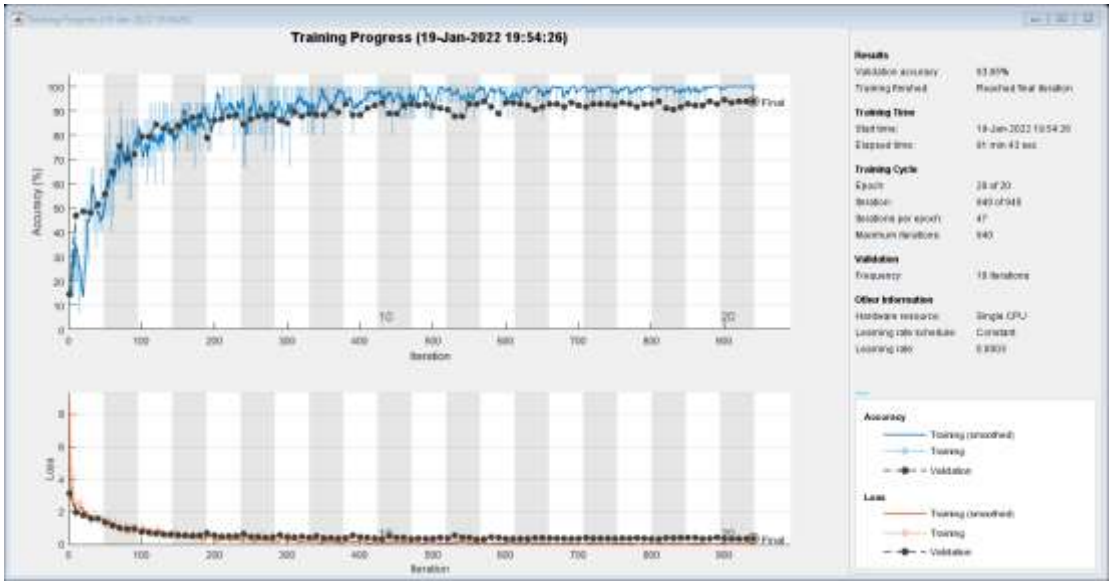


Figure 49 Accuracy and Loss values of classifying 12 arrhythmia classes using GoogleNet and CWT

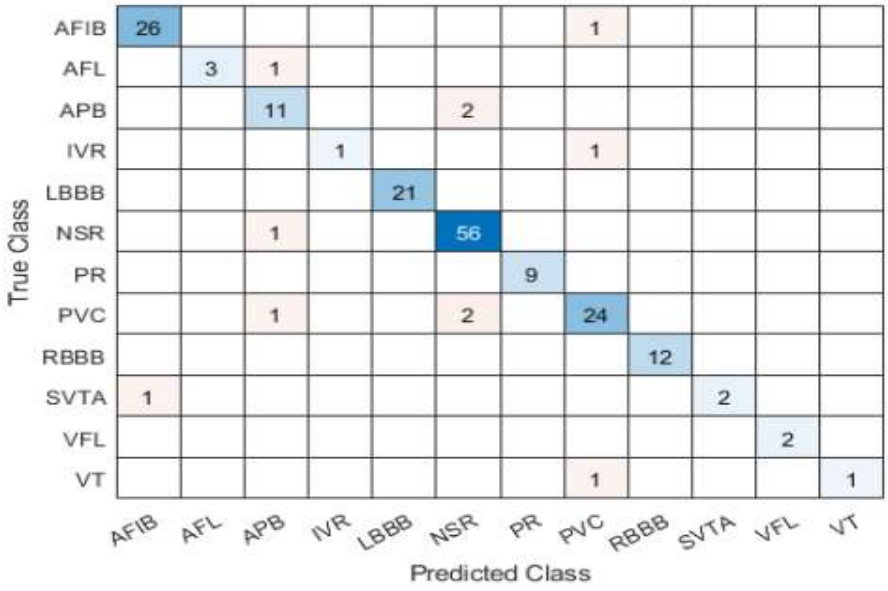


Figure 50 Confusion matrix for classifying 12 arrhythmia using GoogleNet and CWT

Table 9 Precision, sensitivity, specificity,f1-score, and accuracy of classifying 12 arrhythmia using GoogleNet and CWT

Precision (%)	Sensitivity (%)	Specificity (%)	F1-score	Accuracy (%)	Time per Sample (ms)
96.4	84.1	99.4	89.9	93.9	0.179

- **Results Using SqueezeNet and the CWT**

The 2D CNN classifier SqueezeNet was trained to classify the ECG signals scalograms and the classification accuracy was evaluated from the training progress presented in Fig. 51. Figure 52 presents the confusion matrix of classifying the scalograms of 12 groups of arrhythmias using SqueezeNet.

Figure 51 Accuracy and Loss values of classifying 12 arrhythmia using SqueezeNet and CWT

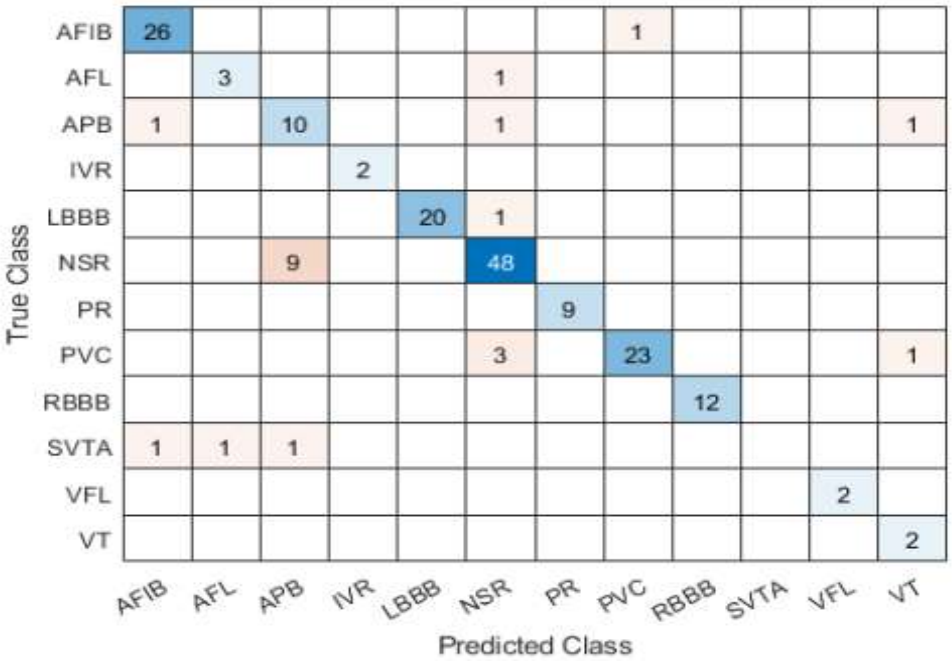


Figure 52 Confusion matrix for classifying 12 arrhythmia using SqueezeNet and CWT

The generated confusion matrix was used to assess the performance of the SqueezeNet network in classifying arrhythmias into twelve classes. Accordingly, the scheme resulted in 79.4% precision, 84.4% sensitivity, 98.8% specificity, F1-score of 81.8 and overall accuracy of 87.7% as also presented in Table 10.

Table 10 Precision, sensitivity, specificity, f1-score, accuracy, and classification time per sample of classifying 12 arrhythmia using SqueezeNet and CWT

Precision (%)	Sensitivity (%)	Specificity (%)	F1-score	Accuracy (%)	Time per Sample (ms)
79.4	84.4	98.8	81.8	87.7	0.152

- **Results Using ResNet-50 and the CWT**

Twelve classes of the generated scalograms were inputted into ResNet-50 to be classified, and the progress of the training was checked as seen in Fig. 53. Figure 54 presents the confusion matrix of classifying the scalograms of 12 groups of arrhythmias using ResNet-50. Accordingly, the scheme resulted in 82.4% precision, 75.1% sensitivity, 99.2% specificity, F1-score of 78.6 and overall accuracy of 91.6% as also presented in Table 11.

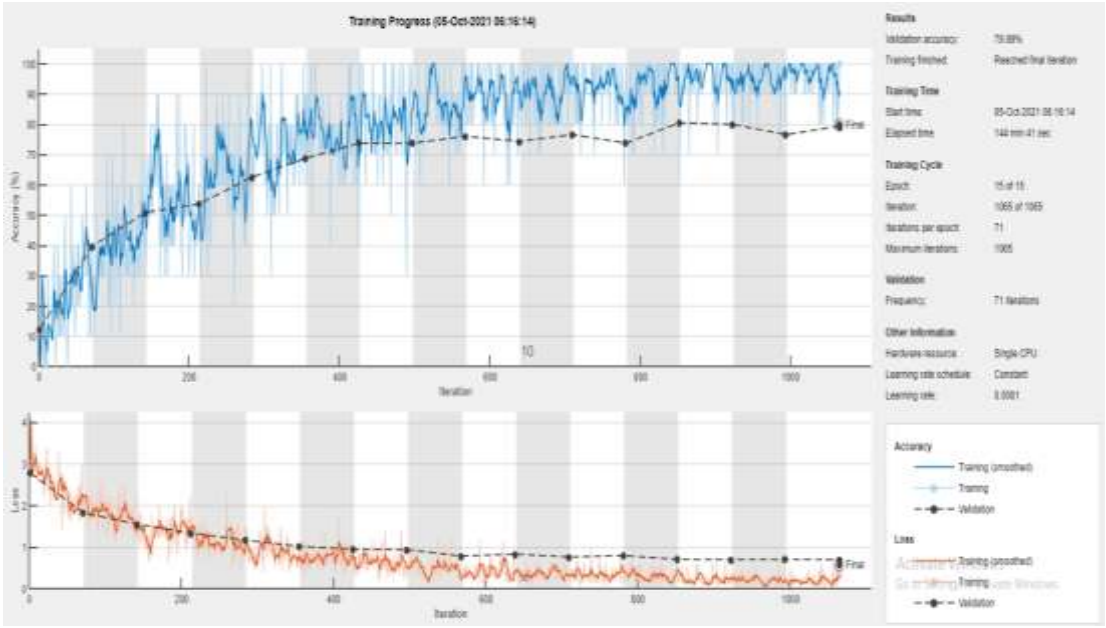


Figure 53 Accuracy and Loss values of classifying 12 arrhythmia using ResNet-50 and CWT

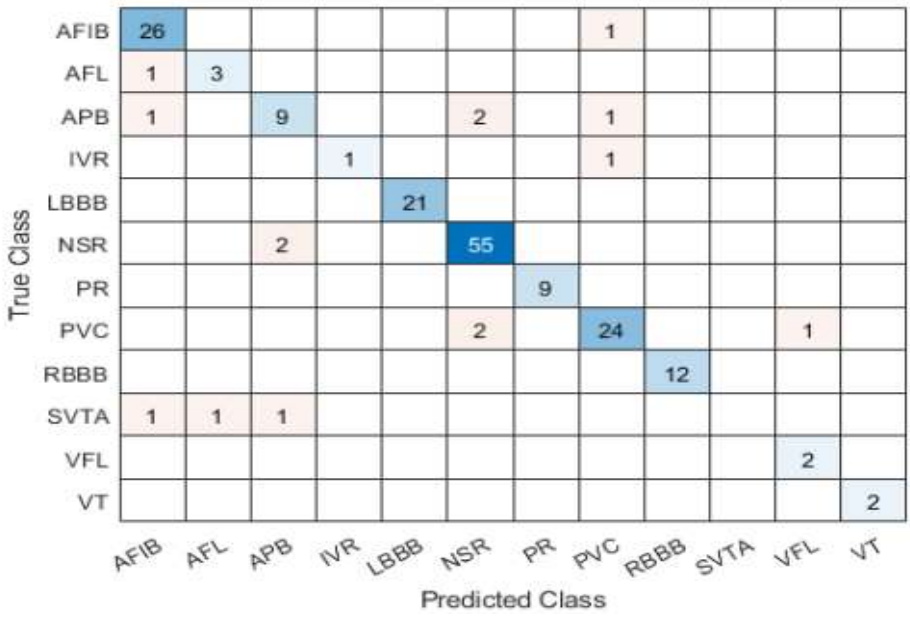


Figure 54 Confusion matrix for classifying 12 arrhythmia using ResNet-50 and CWT

Table 11 Precision, sensitivity, specificity, f1-score, accuracy, and time of classifying 12 arrhythmia using ResNet-50 and CWT

Precision (%)	Sensitivity (%)	Specificity (%)	F1-score	Accuracy (%)	Time per Sample (ms)
82.4	75.1	99.2	78.6	91.6	0.164

C. Classification into 8 Classes

The dataset from MIT-BIH database contains observations that were classified in previous works using different methodologies. The same classes were taken and classified using the proposed methodology: NSR, APB, IVR, LBBB, PR, PVC, RBBB, and VFL. As explained in the methodology section, this test was necessary to make a comparison between the methods proposed in the current study against other methods that were proposed in the literature previously making use of the exact same data set. Figure 55 presents the progress of the training and the computed confusion matrix which allowed computation of the different performance measures. Accordingly, the proposed method offered 97.6% precision, 96.2% sensitivity, 99.4% specificity, F1-score of 96.9, overall accuracy of 96.5% and 0.168 classification time as also been presented in Table 12. This was generated using the GoogleNet architecture applied on the scalograms generated using the CWT.

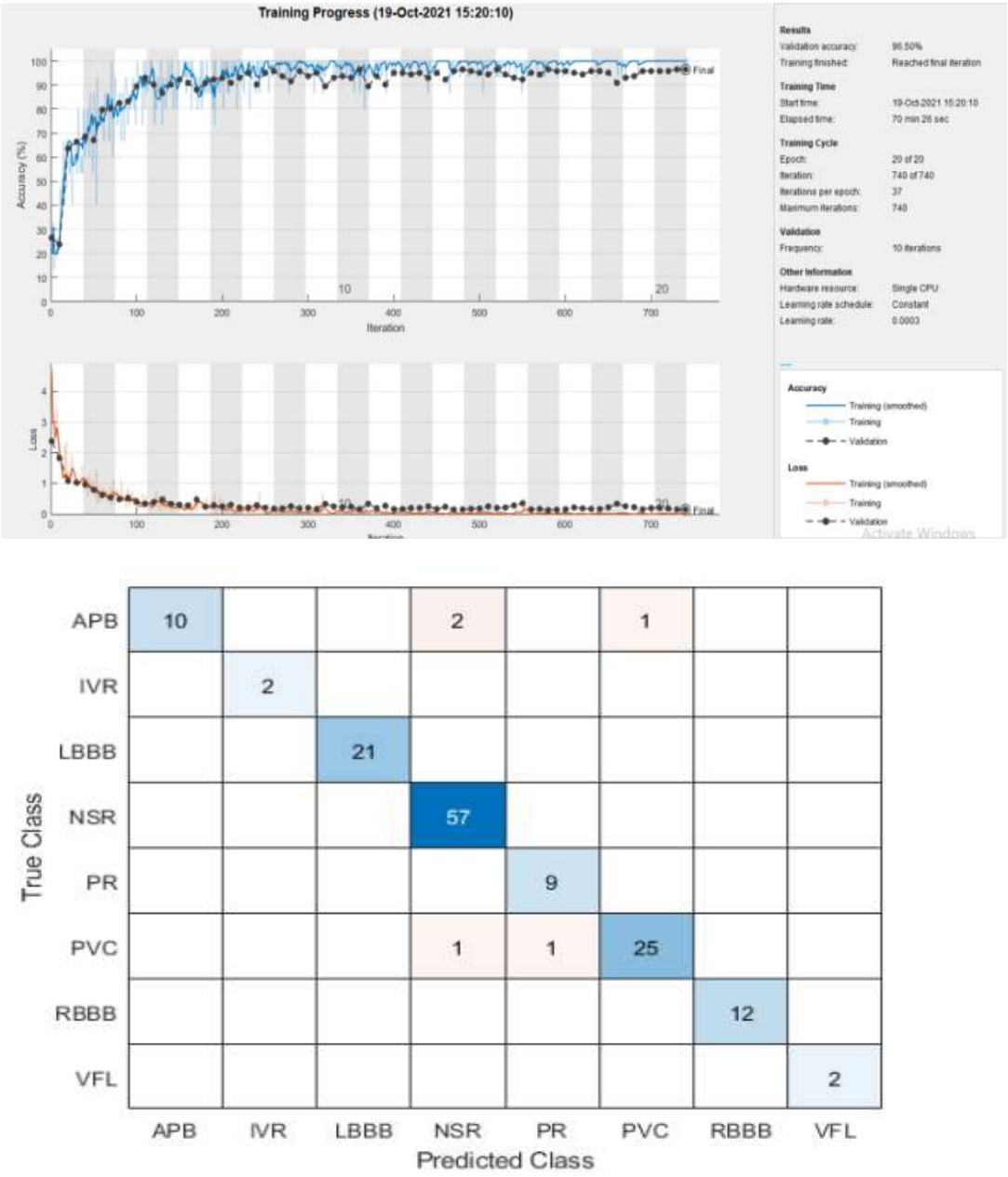


Figure 55 Accuracy and Loss values of classifying eight arrhythmias (top), Confusion matrix (bottom)

Table 12 Precision, sensitivity, specificity, f1-score, and accuracy of classifying 8 arrhythmia

Precision s (%)	Sensitivity (%)	Specificity (%)	F1-score	Accuracy (%)	Time per Sample (ms)
97.6	96.2	99.4	96.9	96.5	0.168

5.3 Discussion

The dataset passed three phases of experimental tests: binary classification and two multiclass classifications. In the binary phase, the dataset were grouped into Normal and Arrhythmia, and the multiclass phase had two sub-phases: eight classes, and twelve classes. In all experiments, useful metrics were calculated to evaluate the model's performance. High

performances were attained in most cases. The best results attained were 98.3% accuracy, 98.3% precision, 97.9% sensitivity, 97.9% specificity, and F1-score of 98.1 for categorizing the data into 2 classes. In the case of 8 group classification, overall accuracy of 96.5%, precision of 97.6%, sensitivity of 96.2%, specificity of 99.4% and F1-score of 96.9 were the best results obtained. In the case of 12-classes, the best accuracy, precision, sensitivity, specificity, F1-score, and classification time were 93.9%, 96.4%, 84.1%, 99.4%, 89.9, and 0.179 ms, respectively. Figure 56 summarizes the best performances of the proposed scheme in terms of the five quantitative matrices for 2, 8 and 12 classes.

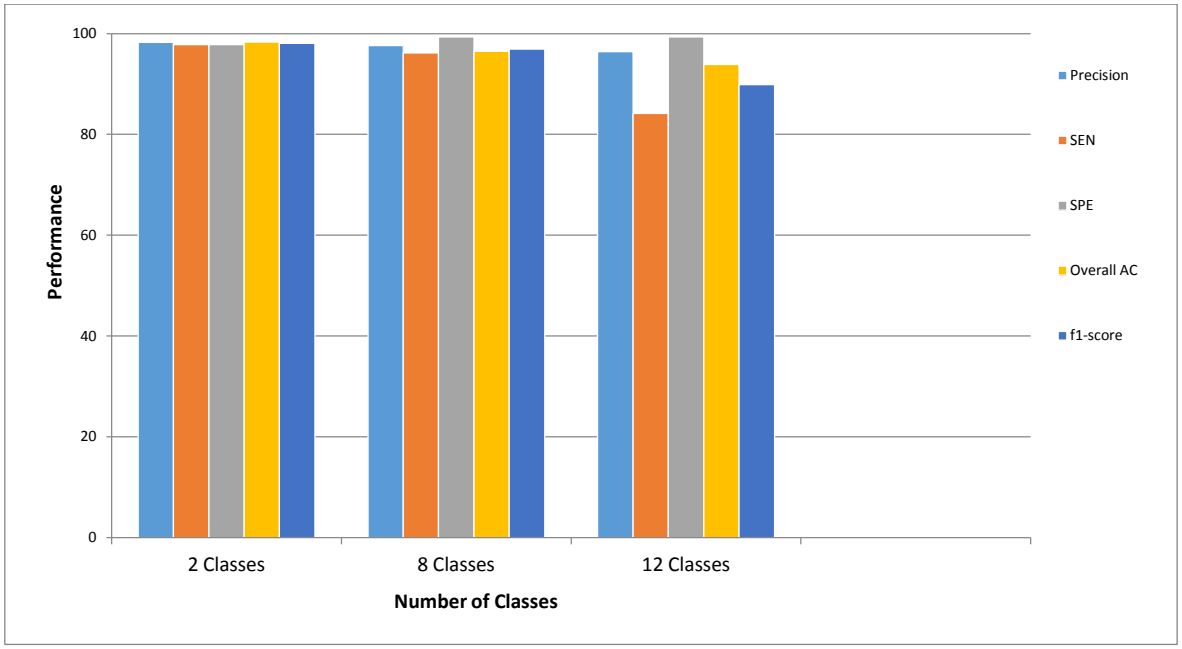


Figure 56 Graphical representation of classification performance for different number of classes

The results present the accuracies achieved for classifying the dataset using several models. The differences in the accuracy between the pretrained models were due to their sensitivity to the training data, and that happened because of the different architectures each of the models made use of. For best trained classifiers, overall accuracy using the CWT stayed in the range of 87.7% - 93.9 %, while using the S-transform the accuracy was between 82.7% and 88.3%. The model that gave the best classification was GoogleNet when using the CWT generated scalograms. ResNet-50 resulted in high accuracy when classifying spectrograms, but it was lower than the performance of GoogleNet with CWT. The results are summarized in Fig. 57.

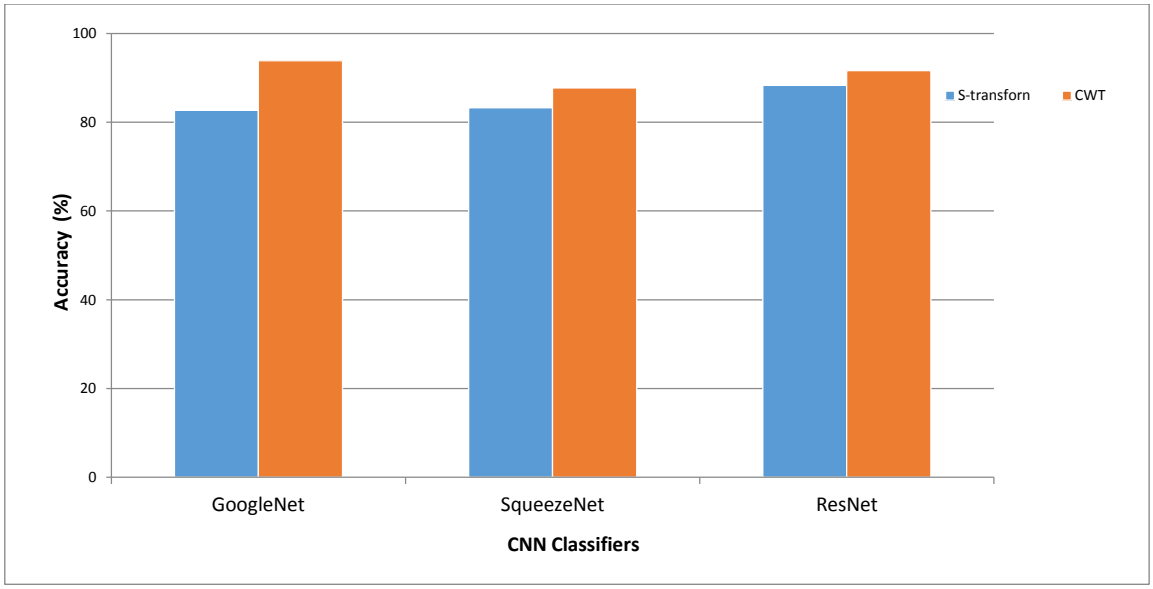


Figure 57 Graphical representation of accuracy using different classifiers with both S-transform and CWT

Based on the results of classifying spectrograms, ResNet-50 showed the best performance with 88.3% accuracy, 85.2% precision, 77.7% sensitivity, 81.3 F1-score and 98.8% specificity. Followed by SqueezNet where the values were 83.2%, 83.3%, 70.4%, 76.3, and 98.2% for accuracy, precision, sensitivity, F1-score and specificity, respectively. GoogleNet offered the lowest performance, though the sensitivity was higher than the one acquired using SqueezeNet. These results are summarized in Fig. 58.

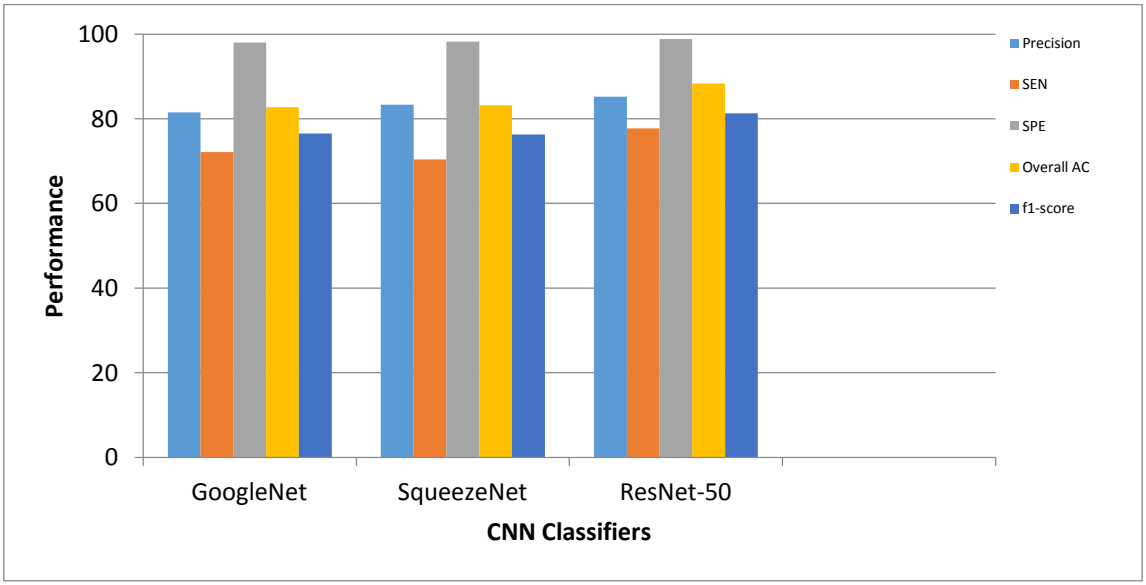


Figure 58 Graphical representation of classification performance using different CNN classifiers with S-transform

Comparing the classification results obtained using the scalograms, GoogleNet showed the best performance with 93.9% accuracy, 96.4% precision, 84.1% sensitivity, 89.9 F1-score,

and 99.4% specificity. The second best model was ResNet-50 where the values were 91.6%, 82.4%, 75.1%, 78.6, and 99.2% for accuracy, precision, sensitivity, F1-score and specificity, respectively. SqueezeNet offered the lowest performance. These results have been summarized in Fig. 59. The differences in performance between the CNN models are attributed to differences in their architecture and the amount of convolution filters used in their structures.

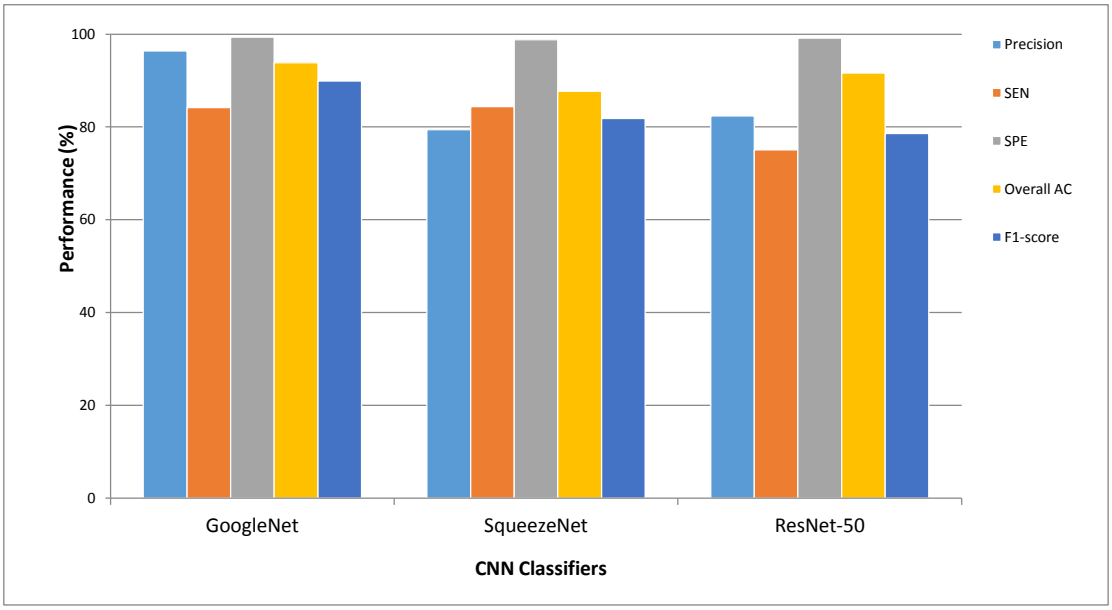


Figure 59 Graphical representation of classification performance using different CNN classifiers with CWT

The computational time needed by all networks to classify scalograms and spectrograms stayed in the range of 0.152 – 0.179 ms. These results are summarized in Fig. 60

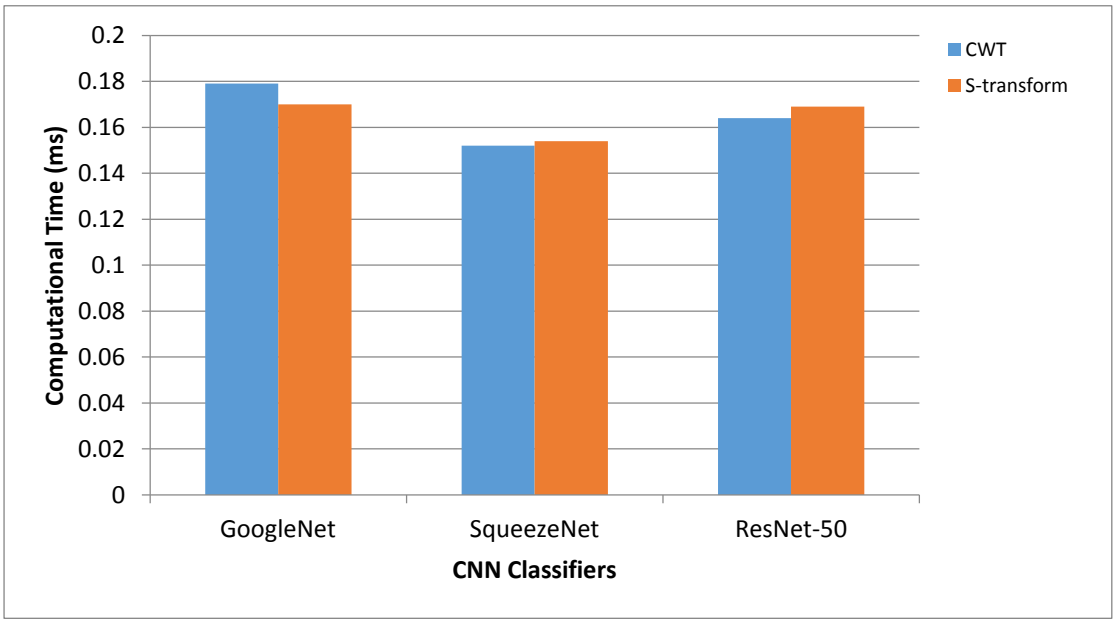


Figure 60 Graphical representation of computational time using different classifiers with both S-transform and CWT

CHAPTER 6

Conclusion and Future Works

6.1 Conclusion

The objective of this study was to find the most suitable 2D CNN classifier that could accurately classify cardiac arrhythmias into twelve diagnostic classifications, including "normal sinus rhythm" and 11 additional rhythm abnormalities using long-duration (10s) ECG signals. In addition, the study assessed the impact of using the CWT and the S-transform in generating images used as inputs for the classifiers.

The proposed method is simple to use and offered high performance for classification of different types of arrhythmias. Training 2D CNN GoogleNet with 0.0003 initial learning rate and 15 mini-batch size achieved the highest accuracy of 93.9% for classifying the time-frequency representation of arrhythmias generated using the CWT into twelve classes. Compared to classifying the images generated using the S-transform, the CWT showed better performance for all three CNN classifiers used. The highest accuracy of 88.3% was achieved when classifying using the S-transform generated images with ResNet-50 model. Compared with recent techniques reported for the classification of twelve different heart beats, the proposed method resulted in higher accuracy, and F1-score.

An overall accuracy of 98.3%, precision of 98.3%, F1-score of 98.1, sensitivity and specificity of 97.9% were obtained when doing binary classification into Normal and Arrhythmia. The same algorithm offered 96.5% accuracy, 97.6% precision, 96.2% sensitivity, 96.9 F1-score, and 99.4% specificity for categorizing the arrhythmias into eight: APB, IVR, LBBB, NSR, PR, PVC, RBBB, and VFL. The proposed scheme attained very close accuracy, sensitivity, specificity, precision (in eight category classification) compared to the result acquired using STFT.

In the current study, the CWT scheme offered superior results compared to the results obtained when using the scheme that makes of the S-transform despite the fact that the S-transform has many interesting properties when dealing with signals like the ECG signals considered in the current study. A validation work on a different data set might reveal interesting comparison between the proposed CWT based and S-transform based schemes.

6.2 Future Works

In order for this work to be clinically applicable for use in arrhythmia classification and help cardiologists in decision making, some improvements might need to be done:

- Testing the performance of the proposed methodology using other dataset of cardiac arrhythmia.
- Use the developed methodology on other physiological signals to evaluate its efficiency.
- Classifying ECG signals acquired from multiple leads;
- Evaluating the system on its efficacy in correctly classifying other types of arrhythmias other than the 11 considered in the current study.

References

[1] https://www.medicalnewstoday.com/articles/257484.

[2] https://www.heart.org/en/health-topics/arrhythmia/about-arrhythmia.

[3] Mc Namara, K.; Alzubaidi, H.; Jackson, J.K, "Cardiovascular disease as a leading cause of death: How are pharmacists getting involved?," 2019.

[4] Gregory Roth, Christopher J.L. Murray, Catherine Johnson, "Global Burden of Cardiovascular Diseases and Risk Factors, 1990–2019," *American College of Cardiology*, 2020.

[5] https://web.archive.org/web/20150303190136/http://www.nhlbi.nih.gov/health/health-topics/topics/arr/atrisk.

[6] https://www.ncbi.nlm.nih.gov/pmc/articles/PMC4064952/.

[7] Rahul Mehra, "Global public health problem of sudden cardiac death," 2007.

[8] Mustaqeem, A.; Anwar, S.M.; Majid, M, "A modular cluster based collaborative recommender system for cardiac patients," 2020.

[9] I Kononenko, "Machine learning for medical diagnosis: History, state of the art and perspective," 2001.

[10] Wooldridge M, Jennings NR, "Intelligent agents: theory and practice," 1995.

[11] Hopfield JJ, "Neural networks and physical systems with emergent collective computational abilities," 1982.

[12] Watts DJ, Strogatz SH, "Collective dynamics of ‘small-world’ networks," 1998.

[13] Ge D, Srinivasan N, Krishnan SM, "Cardiac arrhythmia classification using autoregressive modeling," 2002.

[14] Prasad GK, Sahambi JS, "Classification of ECG arrhythmias using multi-resolution analysis and neural networks," 2003.

[15] https://acsjournals.onlinelibrary.wiley.com/doi/full/10.1002/1097-0142(20010415)91:8+%3C1636:AID-CNCR1176%3E3.0.CO;2-D.

[16] https://towardsdatascience.com/a-comprehensive-guide-to-convolutional-neural-networks-the-eli5-way-3bd2b1164a53.

[17] Roberta Avanzato and Francesco Beritelli, "Automatic ECG Diagnosis Using Convolutional Neural Network," 2020.

[18] Zhang, L., "A Transfer Learning Approach for Handwritten Numeral Digit Recognition," *Hubei University*, 2020.

[19] Awni Y. Hannun, Pranav Rajpurkar, Masoumeh Haghpanahi, Geoffrey H. Tison, Codie Bourn, Mintu P. Turakhia & Andrew Y. Ng , "Cardiologist-level arrhythmia detection and classification in ambulatory electrocardiograms using a deep neural network," 2019.

[20] Anwar, S.M.; Gul, M.; Majid, M.; Alnowami, M, "Arrhythmia Classification of ECG Signals Using Hybrid Features," 2018.

[21] Amin Ullah, Syed Muhammad Anwar, Muhammad Bilal,and Raja Majid Mehmood, "Classification of Arrhythmia by Using Deep Learning with 2-D ECG Spectral Image Representation," 2020.

[22] Lyon, A., Minchole´ , A., Martı´nez, J.P., Laguna, P.,and Rodriguez, B., *Computational techniques for ECG analysis and interpretation in light of their contribution to medical advances*., 2018.

[23] Özal Yıldırım, Paweł Pławiak, Ru-San Tan, U. Rajendra Acharya, "Arrhythmia detection using deep convolutional neural network with long duration ECG signals," 2018.

[24] Robert, P. C.; Mansinha, L., *The S -transform with windows of arbitrary and varying shape*., 2003.

[25] Zhu, Q.; Wang, Y. S.; Shen, G. Q, *Research and comparison of time-frequencytechniques for nonstationary signals*., 2012.

[26] M. Gabriel Khan, "Rapid ECG interpretation, Third edition," , 2003.

[27] R. Acharya, J. S. Suri, J. A.E. Spaan and S.M. Krishnan, "Advances in Cardiac Signal Processing,".

[28] Francis Morris, June Edhouse, William J Brady, John Camm, "ABC of Clinical Electrocardiography," , 2003.

[29] G. Sannino; G. De Pietro, "A deep learning approach for ECG-based heartbeat classification for arrhythmia detection," 2018.

[30] Johns Hopkins Medicine, "The Johns Hopkins Arrhythmia Service," 2018.

[31] Thomas, V, "Premature Ventricular Contractions," *CardioRhythm Management*, 2016.

[32] M. S. Spach, J. M. Kootsey, *The nature of electrical propagation in cardiac muscle*., 1983.

[33] W. Jiang and S. G. Kong, "Block-based neural networks for personalized ECG signal classification," in *IEEE*, 2007.

[34] Mario Sansone, Roberta Fusco, Alessandro Pepino, and Carlo Sansone, *Electrocardiogram Pattern Recognition and Analysis Based on Artificial Neural Networks and Support Vector Machines: A Review*., 2013.

[35] A. J. Moss and S. Stern, "Noninvasive Electro cardiology," , 1996.

[36] https://ecgwaves.com/topic/ekg-ecg-leads-electrodes-systems-limb-chest-precordial/.

[37] https://builtin.com/artificial-intelligence.

[38] Jambukia, S.H., Dabhi, V.K., and Prajapati, H.B, "Classification of ECG signals using machine learning techniques: a survey. 015 International Conference on Advances in Computer Engineering and Applications," 2015.

[39] McCulloch, Warren; Walter Pitts, "A Logical Calculus of Ideas Immanent in Nervous Activity ," 1943.

[40] Dreyfus, Stuart, "The computational solution of optimal control problems with time lag," 1973.

[41] Scarborough D, Somers MJ, "Neural Networks in Organizational Research: Applying Pattern Recognition to the Analysis of Organizational Behaviour," 2006.

[42] A. Bartosch- Härlid B. Andersson U. Aho J. Nilsson R. Andersson, "Artificial neural networks in pancreatic disease," 2008.

[43] Agatonovic-Kustrin S, Beresford R, "Basic concepts of artificial neural network (ANN) modeling and its application in pharmaceutical research," 2000.

[44] da Silva IN, Hernane Spatti S, Andrade Flauzino R, "Chapter 2: Artificial Neural Network Architectures and Training Processes. Artificial Neural Networks: A Practical Course," 2017.

[45] Marshall RC, Freed DB, Karow CM, "Learning of subordinate category names by aphasic subjects: a comparison of deep and surface-level training methods," 2001.

[46] Erhan D, Bengio Y, Courville A, Manzagol PA, Vincent P, Bengio S, "Why does unsupervised pre-training help deep learning?," 2010.

[47] Yuan X, Xie L, Abouelenien M, "A regularized ensemble framework of deep learning for cancer detection from multi-class, imbalanced training data," 2018.

[48] Rodríguez-Pérez R, Bajorath J, "Prediction of compound profiling matrices, part II: relative performance of multitask deep learning and random forest classification on the basis of varying amounts of training data," 2018.

[49] McGarry K, Wermster S, MacIntyre J, "Hybrid Neural Systems: From SImple Coupling to Fully Integrated Nueral Networks. Neural Computing Surveys," 1999.

[50] Kayri M., "Data Optimization with Multilayer Perceptron Neural Network and Using New Pattern in Decision Tree Comparitively," *Journal of Computer Science*, 2010.

[51] Sharma A, Chopra A, "Artificial Neural Networks: Applications in Management," *Journal of Business and Management*, 2013.

[52] Stewart, M., "Intermediate Topics in Neural Networks," 2019.

[53] https://punndeeplearningblog.com/tutorial/overview-of-deep-learning/.

[54] https://towardsdatascience.com/a-comprehensive-guide-to-convolutional-neural-networks-the-eli5-way-3bd2b1164a53.

[55] https://machinelearningmastery.com/convolutional-layers-for-deep-learning-neural-networks/.

[56] https://androidkt.com/how-relu-works-in-convolutional-neural-network/.

[57] https://deepai.org/machine-learning-glossary-and-terms/softmax-layer.

[58] https://en.wikipedia.org/wiki/Softmax_function.

[59] https://www.analyticsvidhya.com/blog/2020/10/how-does-the-gradient-descent-algorithm-work-in-machine-learning/.

[60] https://www.analyticsvidhya.com/blog/2020/10/how-does-the-gradient-descent-algorithm-work-in-machine-learning/.

[61] Ruder, S., "An overview of gradient descent optimization algorithms," 2016.

[62] David Rumelhart; Geoffrey Hinton; Ronald Williams, "Learning representations by back-propagating errors," 1986.

[63] Claesen, M., & Moor, B. D, "Hyperparameter Search in Machine Learning," in *The XI Metaheuristics International Conference*, 2015.

[64] Goodfellow, I., Bengio, Y., & Courville, A, "Deep Learning. USA: www.deeplearningbook.org.," 2017.

[65] Gibson, A., & Patterson, J, "Deep Learning. O'Reilly Media, Inc," 2017.

[66] https://resources.pcb.cadence.com/blog/2020-time-domain-analysis-vs-frequency-domain-analysis-a-guide-and-comparison.

[67] https://en.wikipedia.org/wiki/Frequency_domain.

[68] J. B. J. Fourier, "Théorie analytique de la chaleur," 1822.

[69] https://www.intechopen.com/chapters/74096.

[70] https://en.wikipedia.org/wiki/Fourier_transform.

[71] https://lpsa.swarthmore.edu/Fourier/Xforms/FXformIntro.html.

[72] https://www.sciencedirect.com/topics/computer-science/fourier-transform.

[73] Huang, M.; Yu, P. B., *Application ofsignal analysis in seismic wave processing.*, 2011.

[74] Black, C. J. , *Dynamic Analysis of Civil Engineering Structures Using Joint Time-Frequency Method (Ph.D. Thesis).*, 1998.

[75] Roshan Kumar1, *, Wei Zhao1 and Vikash Singh2, *Joint Time-Frequency Analysis of Seismic Signals: A Critical Review.*, 2018.

[76] https://www.mathworks.com/help/wavelet/ug/time-frequency-analysis-and-continuous-wavelet-transform.html?s_tid=srchtitle_continuous%20wavelet%20transform_3.

[77] Grossmann A, Morlet J, "Decomposition of Hardy Functions into Square Integrable Wavelets of Constant Shape," *SIAM Journal on Mathematical Analysis*, 1984.

[78] https://www.weisang.com/en/documentation/timefreqspectrumalgorithmscwt_en/.

[79] Seo, J.I.; Park, E.K.; Jang, J.Y., "Comparison of ERG denoising performance according to mother function of," 2016.

[80] Olhede, S.C.; Walden, A.T., "Generalized morse wavelets 2002," *IEEE Trans. Signal Process.*, 2002.

[81] Mustaqeem, A.; Anwar, S.M.; Majid, M, "Multiclass classification of cardiac arrhythmia using improved feature selection and SVM invariants," 2018.

[82] Mustaqeem, A.; Anwar, S.M.; Majid, M.; Khan, A.R, "Wrapper method for feature selection to classify cardiac arrhythmia," 2017.

[83] Coast, D.A.; Stern, R.M.M.; Cano, G.G.; Briller, S.A, "An approach to cardiac arrhythmia analysis using hidden markov models," 1990.

[84] Willems, J.L.; Lesaffre, E, "Comparison of multigroup logistic and linear discriminant ecg and vcg classification," 1987.

[85] Dehan, L.; Guanggui, X.U.; Yuhua, Z.; Hosseini, H.G, "Novel ECG diagnosis model based on multi-stage artificial neural networks," 2008.

[86] Ceylan, R.; Ozbay, Y, "Comparison of FCM, PCA and WT techniques for classification ECG arrhythmias using artificial neural network," 2007.

[87] Salvatore, C.; Cerasa, A.; Battista, P.; Gilardi, M.C.; Quattrone, A.; Castiglioni, I, "Magnetic resonance imaging biomarkers for the early diagnosis of Alzheimer's disease: A machine learning approach," 2015.

[88] Kiranyaz, S.; Ince, T.; Gabbouj, M, "Real-time patient-specific ECG classification by 1-D convolutional neural networks," 2015.

[89] Rajpurkar, P.; Hannun, A.Y.; Haghpanahi, M.; Bourn, C.; Ng, A.Y, "Cardiologist-level arrhythmia detection with convolutional neural networks," 2017.

[90] Acharya, U.R.; Oh, S.L.; Hagiwara, Y.; Tan, J.H.; Adam, M.; Gertych, A.; San Tan, R, "A deep convolutional neural network model to classify heartbeats," 2017.

[91] R. Salloum and C.-C. J. Kuo, "'ECG-based biometrics using recurrent neural networks," 2017.

[92] A. Mostayed, J. Luo, X. Shu, and W. Wee, "Classification of 12-lead ECG signals with Bi-directional LSTM network," 2018.

[93] C. Zhang, G. Wang, J. Zhao, P. Gao, J. Lin, and H. Yang, "Patient-specific ECG classification based on recurrent neural networks and clustering technique," 2017.

[94] Isin, A. and Ozdalili, S., "Cardiac arrhythmia detection using deep learning," *Procedia Computer Science*, 2017.

[95] Jinkwon Kim, Hang Sik Shin, Kwangsoo Shin and Myoungho Lee, "Robust algorithm for arrhythmia classification in ECG using extreme learning machine," 2009.

[96] Brijesh Verma , Syed Zahid Hassan , "Hybrid ensemble approach for classification," 2009.

[97] Francisco J. Veredas, Héctor Mesa, Laura Morente , "A hybrid learning approach to tissue recognition in wound images," 2009.

[98] B. Samanta, C. Nataraj, "Automated diagnosis of cardiac state in healthcare systems using computational intelligence," 2008.

[99] Mohsen Sangi ,Khin Than Win,Farid Shirvani,Mohammad-Reza Namazi-Rad,Nagesh Shukla, "Applying a Novel Combination of Techniques to Develop a Predictive Model for Diabetes Complications," 2015.

[100] Wong TC, Chin KS Xu M, "Modeling daily patient arrivals at Emergency Department and quantifying the relative importance of contributing variables using artificial neural network," 2013.

[101] Su¨ t N, Senocak M, "Assessment of the performances of multilayer perceptron neural networks in comparison with recurrent neural networks and two statistical methods for diagnosing coronary artery disease," 2007.

[102] Fábio S. Aguiar, Rodrigo C. Torres, João V. F. Pinto, Afrânio L. Kritski, José M. Seixas & Fernanda C. Q. Mello , "Development of two artificial neural network models to support the diagnosis of pulmonary tuberculosis in hospitalized patients in Rio de Janeiro, Brazil," 2016.

[103] Mohanty, M.D.; Mohanty, B.; Mohanty, M.N, "R-peak detection using efficient technique for tachycardia detection," in *In Proceedings of the 2017 2nd International Conference on Man and Machine Interfacing (MAMI)*, 2017.

[104] Gu, J.; Wang, Z.; Kuen, J.; Ma, L.; Shahroudy, A.; Shuai, B.; Liu, T.; Wang, X.; Wang, G.; Cai, J.; et al, "Recent advances in convolutional neural networks," 2018.

[105] Wu, Y.; Yang, F.; Liu, Y.; Zha, X.; Yuan, S, "comparison of 1-D and 2-D deep

convolutional neural networks in ECG classification," 2018.

[106] Urtnasan Erdenebayar, Hyeonggon Kim, Jong-Uk Park Dongwon Kang, and Kyoung-Joung Lee, "Automatic Prediction of Atrial Fibrillation Based on Convolutional Neural Network Using a Short-term Normal Electrocardiogram Signal," 2019.

[107] Dan Li, Jianxin Zhang*, Qiang Zhang*, Xiaopeng Wei, "Classification of ECG Signals Based on 1D Convolution Neural Network," 2017.

[108] U. Rajendra Acharya, Shu Lih Oh, Yuki Hagiwara, Jen Hong Tan, Muhammad Adam,Arkadiusz Gertych, Tan Ru San, "A deep convolutional neural network model to classify heartbeats," 2017.

[109] Goldberger AL, Amaral LAN, Glass L, Hausdorff JM, Ivanov PCh, Mark RG, Mietus JE, Moody GB, Peng C-K, Stanley HE, "PhysioBank, PhysioToolkit, and PhysioNet: Components of a New Research Resource for Complex Physiologic Signals," 2003.

[110] Lilly, J.M.; Olhede, S.C., "Higher-order properties of analytic wavelets," *IEEE Trans. Signal Process*, 2009.

[111] Lilly, J.M.; Olhede, S.C. , "Generalized Morse wavelets as a superfamily of analytic wavelets.," *IEEE Trans.Signal Process*, 2012.

[112] Yeong-Hyeon Byeon, Sung-Bum Pan and Keun-Chang Kwak, "Intelligent Deep Models Based on Scalograms of Electrocardiogram Signals for Biometrics," 2019.

[113] Paul S Addison, "Wavelet transforms and the ECG: a review," 2005.

[114] https://paperswithcode.com/method/googlenet#.

[115] https://paperswithcode.com/method/inception-module.

[116] https://blog.paperspace.com/popular-deep-learning-architectures-resnet-inceptionv3-squeezenet/.

[117] Abrol, A., Bhattarai, M., Fedorov, A., Du, Y., Plis, S. and Calhoun, V., "Deep residual learning for neuroimaging: An application to predict progression to Alzheimer's disease," *Journal of Neuroscience Methods*, 2020.

[118] W. Yin, X. Yang, L. Zhang, and E. Oki, "'ECG monitoring system integrated with IR-UWB radar based on CNN," *IEEE* , 2016.

[119] Fei Jiang, Yong Jiang, Hui Zhi, Yi Dong, Hao Li, Sufeng Ma, Yilong Wang, Qiang Dong, Haipeng Shen, Yongjun Wang, "Artificial intelligence in healthcare: past, present and future ," 2017.

[120] Paulo J.Lisboa, Azzam F.G.Taktak, "The use of artificial neural networks in decision support in cancer: A systematic review," 2006.

[121] Paliwal M, Kumar U, "Neural networks and statistical techniques: A review of

applications. Expert Systems with applications," 2009.

[122] Tu JV, "Advantages and disadvantages of using artificial neural networks versus logistic regression for predicting medical outcomes," 1996.

[123] Frederic Stahl, Ivan Jordanov, "An overview of the use of neural networks for data mining tasks," 2012.

[124] Pei J, Kamber M Han J, "Data Mining: Concepts and Techniques," 2012.

[125] Picard RR, Ber KN., "Data Splitting," 2010.

[126] Reitermanova Z., "Data Splitting. WDS'10 Proceedings of Contributed Papers," 2010.

[127] R.J.May, H.R.Maier, G.C.Dandy, "Data splitting for artificial neural networks using SOM-based stratified sampling," 2010.

[128] Amazon, "Amazon Machine Learning: Developer Guide," 2018.

[129] N. ChSriman Narayana Iyengar T.Vivekanandan, "Optimal feature selection using a modified differential evolution algorithm and its effectiveness for prediction of heart disease," 2017.

[130] Olatubosun O, Olusoga F, Abayomi F, "Diabetes Diagnosis with Maximum Covariance Weighted Resilience," *International Journal of Health and Economic Development*, 2015.

[131] F. Ibrahim, T. Faisal, M. I. Mohamad Salim & M. N. Taib , "Non-invasive diagnosis of risk in dengue patients using bioelectrical impedance analysis and artificial neural network," 2010.

[132] Anderson I, Maitland J, Sherwood S, Barkhuss L, Chalmers M, Hall M, et al, "Shakra: Tracking and Sharing Daily Activity Levels with Unaugmented Mobile Phones. Mobile Networks and Applications," 2007.

[133] Stephan Kudyba, Thomas Gregorio, "Identifying factors that impact patient length of stay metrics for healthcare providers with advanced analytics," 2011.

[134] Kononenko I, "Machine learning for medical diagnosis: history, state of the art and perspective," 2001.

[135] Minsky M, "Steps toward artificial intelligence," 1961.

[136] Weng J, McClelland J, Pentland A, Sporns O, Stockman I, Sur M, et al, "Autonomous mental development by robots and animals," 2001.

[137] LeCun Y, Bengio Y, Hinton G, "Deep learning," 2015.

[138] Rose DC, Karnowski TP Arel I, "Deep machine learning- a new frontier in artificial intelligence research," 2010.

[139] Dudek-Dyduch E, Tadeusiewicz R, Horzyk A, "Neural network adaptation process effectiveness dependent of constant training data availability," 2009.

[140] Guoguang Rong, Arnaldo Mendez, Elie Bou Assi, Bo Zhao, Mohamad Sawan, "Artificial Intelligence in Healthcare: Review and Prediction Case Studies," 2020.

[141] Elkin PL, Schlegel DR, Anderson M, Komm J, Ficheur G, Bisson L, "Artificial intelligence: bayesian versus heuristic method for diagnostic decision support," 2018.

[142] Safdar S, Zafar S, Zafar N, Khan NF, "Machine learning based decision support systems (DSS) for heart disease diagnosis: a review," 2018.

[143] Friesen GM, Jannett TC, Jadallah MA, Yates SL, Quint SR, "A Comparison of the Noise Sensitivity of Nine QRS Detection Algorithms," 1990.

[144] Ceylan, R.; Ozbay, Y, "Comparison of FCM, PCA and WT techniques for classification ECG arrhythmias using artificial neural network," 2007.

[145] Polat, K.; Günes, S, "Breast cancer diagnosis using least square support vector machine," 2007.

[146] Bashir, A.K.; Arul, R.; Basheer, S.; Raja, G.; Jayaraman, R.; Qureshi, N.M.F, "An optimal multitier resource allocation of cloud RAN in 5G using machine learning," 2019.

[147] Lackland, D.T.; Weber, S.M.A, "Global burden of cardiovascular disease and stroke: hypertension at the core," 2015.

[148] Minami, K.I.; Nakajima, H.; Toyoshima, T, "Real-time discrimination of ventricular tachyarrhythmia with Fourier-transform neural network," 1999.

[149] Hu, Y.H.; Palreddy, S.; Tompkins,W.J, "A patient-adaptable ECG beat classifier using a mixture of experts approach," 1997.

[150] A. Ecar, "Recommended practice for testing and reporting performance results of ventricular arrhythmia detection algorithms," 1987.

[151] Nida Shahid, Tim Rappon, Whitney Berta, "Applications of artificial neural networks in health care organizational decision-making: A scoping review," 2019.

[152] Peter B.Snow, Robert J.Almassy, William J.Oetgen, "Artificial neural networks: Current status in cardiovascular medicine," 1996.

[153] Clevert, D.A.; Unterthiner, T.; Hochreiter, S, "Fast and accurate deep network learning by exponential linear units (elus)," 2015.

[154] Moody GB, Mark RG, "The impact of the MIT-BIH arrhythmia database," , 2001.

[155] N. WILSON, CHARLES E. K, GEORGE E. B, EMANUEL. G, ASHTON.G, HANS H., FRANKLIND. , E. L, and GORDON B. M, *Recommendations for Standardization*

of Electrocardiographic and Vector-cardiographic Leads., 1954.

[156] Qin, Q., Li, J., Zhang, L., Yue, Y., and Liu, C., *Combining low-dimensional wavelet features and support vector machine for arrhythmia beat classification*., 2017.

[157] R. Begg, *Neural Networks in Healthcare:Potential and Challenges: Potential and Challenges (IGI Global)*., 2006.

[158] Tsai-Min Chen,Chih-Han Huang,Edward S.C. Shih,Yu-Feng Hu,and Ming-Jing Hwang, *Detection and Classification of Cardiac Arrhythmias by a Challenge-Best Deep Learning Neural Network Model*., 2020.

[159] Özal Yıldırım, Paweł Pławiak, Ru-San Tan, U. Rajendra Acharya, *Arrhythmia detection using deep convolutional neural network with long duration ECG signals*., 2018.

[160] JINGSHAN HUANG , BINQIANG CHEN, BIN YAO, AND WANGPENG HE , *ECG Arrhythmia Classification Using STFT-Based Spectrogram and Convolutional Neural Network*., 2019.

[161] Li, Y. D.; Zheng, X., *Wigner-Ville distribution and its application in seismic attenuation estimation*., 2007.

[162] Amy, N. R.; Wiesław, J. S., *Time-frequency and time-scale analyses for structural health monitoring*., 2007.

[163] Christopher, W. L.; Kyle, F., *Time-frequency analysis with the continuous wavelet transform*., 1998.

[164] Box G. E. P. and Jenkins G., *Time Series Analysis: Forecasting and Control*., 1976.

[165] R.G. Stockwell, *A basis for efficient representation of the S-transform*., 2006.

[166] Ting Shu, Bob Zhang, and Yuan Yan Tang, "Effective Heart Disease Detection Based on Quantitative Computerized Traditional Chinese Medicine Using Representation Based Classifiers," 2017.

[167] Hugo Pl´acido da Silva, Carlos Carreiras, Andr´e Louren¸co, Ana Fred,Rui C´esar das Neves, and Rui Ferreira, "Off-the-person electrocardiography: performance assessment and clinical correlation," 2015.

[168] S. Osowski, T. H. Linh, and T. Markiewicz, "Support vector machinebased expert system for reliable heart beat recognition," , 2004.

[169] P.de Chazal, M.O. Duyer, and R.B. Reilly, "“Automatic classification of heartbeat using ECG morphology and heart beat interval features," , 2004.

[170] Omern T. Inan. L. Giovangrandi, and T. A. Kovacs, "“Robust Neural network based classification of Premature Ventricular Contraction using wavelet transform and time interval features," , 2006.

[171] Y. Hu, S. Palreddy, and W. J. Tompkins, "A patient-adaptable ECG beat classifier using a mixture of experts approach," , 1997.

[172] S. Osowski, T. H. Linh, and T. Markiewicz, "Support vector machine based expert system for reliable heart beat recognition," , 2004.

[173] (1998) http://www.physionet.org/physiobank/database/mitdb/.

[174] R. G. Stockwell, L. Mansinha, and R. P. Lowe, "Localization of the Complex Spectrum: The S Transform," 1996.

[175] https://en.wikipedia.org/wiki/Short-time_Fourier_transform.

[176] https://ebn.bmj.com/content/23/1/2.

[177] https://en.wikipedia.org/wiki/Accuracy_and_precision.

[178] Tsai-Min Chen, Chih-Han Huang, Edward S.C. Shih, Yu-Feng Hu,and Ming-Jing Hwang, "Detection and Classification of Cardiac Arrhythmias by a Challenge-Best Deep Learning Neural Network Model," 2020.

[179] https://www.nibib.nih.gov/science-education/science-topics/artificial-intelligence-ai.

[180] Muayed S Al-Huseiny, Noor Khudhair Abbas, Ahmed S Sajit, "Diagnosis of arrhythmia based on ECG analysis using CNN," 2020.

[181] Nordqvist, C, "Arrhythmia: Causes, symptoms, types, and treatment," 2017.

[182] Awni Y. Hannun, Pranav Rajpurkar, Masoumeh Haghpanahi, Geoffrey H. Tison, Codie Bourn, "Cardiologist-level arrhythmia detection and classification in ambulatory electrocardiograms using a deep neural network," 2019.

[183] Johns Hopkins Medicine, "The John Hopkins Arrhythmia Service," 2018.

[184] Özal Yıldırım, Paweł Pławiak, Ru-San Tan, U. Rajendra Acharya, "Arrhythmia detection using deep convolutional neural network with long duration ECG signals," 2018.

[185] Li, T.; Zhou, M., "ECG classification using wavelet packet entropy and random forests," 2016.

[186] (2018) jLab: A Data Analysis Package for Matlab. Available online: http://www.jmlilly.net/jmlsoft.html.

[187] Paul S Addison, "Wavelet transforms and the ECG: a review".