Spring 2019

Automated Atrial Fibrillation Detection from Electrocardiogram

Senbao Lu

SANTA CLARA UNIVERSITY

Department of Bioengineering

I HEREBY RECOMMEND THAT THE THESIS PREPARED
UNDER MY SUPERVISION BY

Senbao Lu

ENTITLED

Automated Atrial Fibrillation Detection from Electrocardiography

BE ACCEPTED IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR THE DEGREE OF

MASTER'S OF SCIENCE
IN
BIOENGINEERING

06/17/2019

Thesis Advisor: Dr. Yuling Yan date

06/17/2019

Department Chair: Dr. Zhiwen Zhang date

Automated Atrial Fibrillation Detection from Electrocardiogram

by

Senbao Lu

Master's Thesis

Submitted to
the Department of Bioengineering

of

SANTA CLARA UNIVERSITY

in partial fulfillment of the requirements
for the degree of
Master's of Science in Bioengineering

Santa Clara, California

Spring 2019

Acknowledgements

I would like to first thank Dr. Yuling Yan for her bounteous help as my research mentor and thesis advisor during my five years' study in Bioengineering Department of Santa Clara University. Without her help, I would not be able to find my passion in computational biology and bioinformatics and continue further study in this area.

I would also like to thank all faculty and staff members in BIOE Department, especially Dr. Bill Lu as my undergraduate academic advisor. I learned so much knowledge and skills from all classes in both academia and industry and I believe I will utilize things I learned in Santa Clara University for my future study and work.

Finally, I would like to thank my family for supporting me to study abroad. Your love and support make it possible for me to chase my dream across Pacific Ocean. I hope you guys are proud of me.

Automated Atrial Fibrillation Detection from Electrocardiogram

Senbao Lu

Department of Bioengineering
Santa Clara University
June 17, 2019

ABSTRACT

In this study, a novel Atrial Fibrillation (AFib) detection algorithm is presented based on Electrocardiography (ECG) signals. In particular, the spectrogram of ECG signal is used as an input to a Convolutional Neural Network (CNN) to classify normal and AFib ECG signals. This model is shown to perform well with an accuracy of 92.91% and a value of 0.9789 for the area under the ROC curve (AUC). This study demonstrated the potential of using image classification methods and CNN model to detect abnormal biosignals with noise.

Contents

1 Introduction **1**
1.1 Background and Motivation 1
1.2 Current Technology and Problems 2
1.3 Goals and Contributions 3

2 System Level Overview **4**
2.1 Data Source 4
2.2 Architecture 5
2.3 Expected Result 6

3 Methods **7**
3.1 Filter 7
3.1.1 Introduction 7
3.1.2 Options 8
3.1.3 Design Description 8
3.1.4 Analysis and Test Result 10
3.2 Hilbert Diagram 10
3.2.1 Introduction 10
3.2.2 Options 11
3.2.3 Design Description 11
3.2.4 Analysis and Test Result 12
3.3 Spectrogram 12
3.3.1 Introduction 12
3.3.2 Options 13
3.3.3 Design Description 13
3.3.4 Analysis and Test Result 13
3.4 CNN 14
3.4.1 Introduction 14
3.4.2 Options 15
3.4.3 Design Description 15

3.4.4 Analysis and Test Result . . . 16
3.5 Transfer Learning . . . 18
3.5.1 Introduction . . . 18
3.5.2 Options . . . 18
3.5.3 Design Description . . . 18
3.5.4 Analysis and Test Result . . . 19
3.6 Data Augmentation . . . 20
3.6.1 introduction . . . 20
3.6.2 Options . . . 21
3.6.3 Design Description . . . 21
3.6.4 Analysis and Test Result . . . 21

4 Results **23**
4.1 Protocol . . . 23
4.1.1 Training . . . 23
4.1.2 Testing . . . 24
4.2 Results . . . 24
4.3 Discussion . . . 25

5 Conclusion **27**

List of Figures

2.1 Sample signals of both normal (top) and AFib (bottom). 5

3.1 Fourier Spectrum of a sample normal signal. The red boxes represent the heartbeats and their harmonics and the blue boxes represent low frequency noise. 9
3.2 Fourier Spectrum of a sample AFib signal. The red boxes represent the heartbeats and the blue boxes represent low frequency noise. . . 9
3.3 Sample signals filtered by bandpass filter. 10
3.4 A phase plot of normal (left) and AFib (right) ECG signals 12
3.5 A sample spectrogram of normal ECG 14
3.6 A sample spectrogram of AFib ECG 15
3.7 A simplified CNN layer graph 16
3.8 The classification result for Hilbert phase plot using a simple CNN . 17
3.9 The classification result for spectrogram using a simple CNN 17
3.10 A detailed structure of GoogLeNet Inception 1 19
3.11 The classification result for GoogLeNet Inception 1 20
3.12 A sample classification output for GoogLeNet Inception 1. Nnew stands for normal and A stands for AFib. The percentage next the Nnew or A is the confidence of making the decision 20
3.13 The classification result for GoogLeNet Inception 1 with data augmentation 22
3.14 A sample classification output for GoogLeNet Inception 1 with data augmentation. Nnew stands for normal and A stands for AFib. The percentage next the Nnew or A is the confidence of making the decision 22

4.1 The training progress plot for classification 25
4.2 The ROC curve for classification 25

Glossary of Terms

1. Biological

1. AFib: Atrial Fibrillation.
2. ECG: Electrocardiogram.
3. AV: AtrioVentricular

2. Signal Processing

1. CNN: Convolutional Neural Network.
2. FFT: Fast Fourier Transform.
3. MATLAB: A multi-paradigm numerical computing environment and proprietary programming language developed by MathWorks.
4. IIR: Infinite Impulse Response.
5. GPU: Graphics Processing Unit.
6. GeForce GTX 1080: A GPU with pascal architecture, 8 GB GDDR5X Frame Buffer, and 10GBps memory speed.
7. ROC curve: Receiver Operating Characteristic curve.
8. AUROC: Area Under ROC curve.

Chapter 1

Introduction

1.1 Background and Motivation

Computational Biology is a research area that uses data generated from previous biological experiments and computational models to find patterns of biological systems. Some common biological data used in computational biology research includes biosignals, neuroimages, and genomics and proteomics data. The main computational models used in such research are statistical machine learning models and deep learning models, thanks to the huge advance in Artificial Intelligence research. Statistical machine learning techniques like Support Vector Machine (SVM) and decision trees are widely used in biosignal processing and pattern recognition in genomics. Deep learning is a set of machine learning methods based on artificial neural networks that use multiple layers to extract high-level features from raw input. It is extensively used in computer vision and natural language processing, which makes it a good candidate for computational biology research in bioimage processing and pattern recognition in genomics and proteomics.

Atrial Fibrillation (AFib) is a supraventricular tachyarrhythmia with uncoordinated atrial activation and consequently ineffective atrial contraction [1] and is the most common of the serious cardiac rhythm disturbances [2]. AFib has a significant impact on longevity, increasing all-cause and cardiovascular mortality rates [3] [4] and can lead to blood clots, stroke [5], heart failure and other heart-related complications. It is estimated that 2.7 to 6.1 million people in the United States have AFib with an expectation to increase, where 2% of people under age of 65 and 9% of people above age of 65 have AFib [1]. AFib costs the United States about 6 billion dollar each year and 8,705 dollar more for each AFib patient than other patients without AFib [1] [6].

With more and more biological data and health record generated everyday, there is a rising need to properly analyze these data and find hidden patterns that could potentially provide more insight in human diseases and help design more diagnosis tools and treatment options for complex diseases. AFib is among one of such diseases. Over 10% patients with hypertension but no AFib history were detected atrial tachyarrhythmias, which were associated with an increased risk of clinical atrial fibrillation [7]. Early detection of AFib could reduce the medical cost and even save life.

1.2 Current Technology and Problems

In the current medical practice, an AFib is determined by a medical doctor from a 12-lead Electrocardiogram (ECG) graph with the patterns of irregular R-R intervals (when atrioventricular (AV) conduction is present), absence of distinct repeating P waves, and irregular atrial activity [1].

However, many work has been done for automatic detection of AFib from ECG signal without the presence of doctors, and some of such algorithms are embedded in pacemakers to detect AFib events. Verberk, et.al showed a 0.98 sensitivity and 0.92 specificity of AFib detection using blood pressure monitor [8]. Rincon, et.al show a 0.96 sensitivity and 0.93 specificity using a real-time detection method from a wearable wireless sensor platform [9]. Hong, et.al achieved 0.84 F1 score using expert features and Deep Neural Networks [10]. All of these methods show promising results, but a deeper and higher-level features from ECG signals are not used in their detection methods.

1.3 Goals and Contributions

In this study, a novel AFib detection algorithm is developed. It combines signal processing and computer vision techniques to achieve high detection accuracy. Specifically, the spectrogram of ECG signals and a Convolutional Neural Network (CNN) are used to predict whether an ECG signal is from an AFib patient or a normal patient. The contribution of this study is to provide a useful algorithm for AFib detection as well as to show the potential of using image classification methods to reveal deep features in signal processing and classification tasks.

Chapter 2

System Level Overview

2.1 Data Source

As a supervised machine learning project, the accuracy of data label is very important to train a model with high accuracy. In the field of computational biology, it is widely believed that labels made by professional medical doctors are golden standard to compare different models, as we are at the stage where machine learning models serve as a secondary opinion to help doctors make accurate judgment on diseases. For the success of this research, picking a reliable data and label source is the critical first step.

In this study, I use an ECG data package for AFib Classification from PhysioNet Challenge 2017 [11]. PhysioNet is an NIH research resource for complex physiologic signals and is supported by National Institute of General Medical Sciences (NIGMS) and National Institute of Biomedical Imaging and Bioengineering (NIBIB). The ECG data is collected by a single-channel ECG device and is stored as 300 Hz, 16-bit data with bandwidth 0.5 to 40 Hz with +/- 5 mV dynamic range. I use the publicly available training set, which contains 8,528 ECG recordings ranged from 9 seconds to over 60 seconds. I will utilize the 5,076 normal samples and 758

AFib samples for training and testing purpose, ignoring other irrelevant data entries. The label is provided from one expert in this field and I consider it as the ground truth in the scope of this research.

Sample signals of both normal and AFib patients are plotted in Fig.2.1. The normal ECG, despite some noise in the first 10 seconds, has almost constant distance between the spikes, which represents the R-R interval. The AFib ECG, on the other hand, has inconstant R-R interval, which is a significant characteristics of AFib.

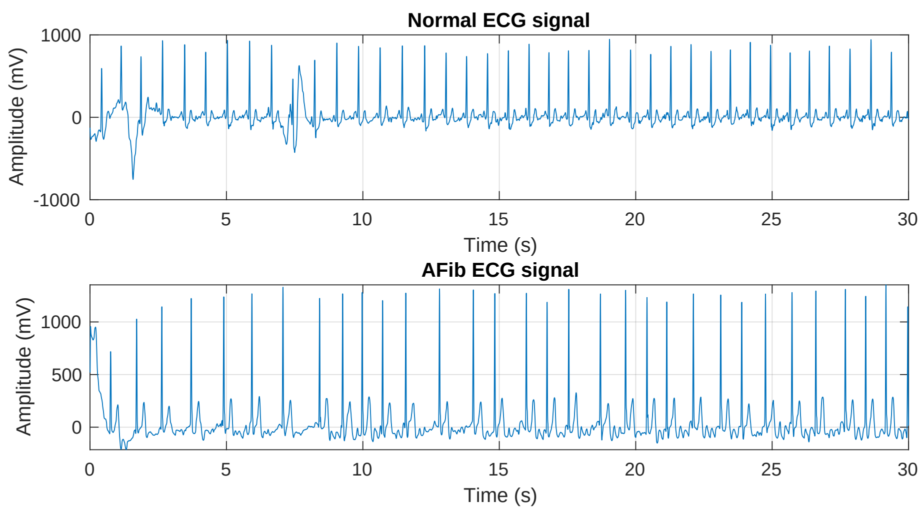

Figure 2.1: Sample signals of both normal (top) and AFib (bottom).

2.2 Architecture

This study is divided into three steps. The first step is to use different signal processing techniques to pre-process the ECG signals from PhysioNet database. This pre-processing process cancels noise from different sources and prepares the signal for further analysis. The second step is to convert ECG signals into 2-dimensional images with information from both time and frequency domain. This step reveals ECG patterns in both domain simultaneously and makes it possible to use computer vision and image processing techniques to perform further analysis. The last step is to use Convolutional Neural Network (CNN) to find higher level patterns of AFib from time and frequency domain in ECG signal and perform classification tasks.

2.3 Expected Result

The primary goal for this study is to achieve high accuracy on AFib detection. A secondary goal is to achieve high efficiency and robustness

Chapter 3

Methods

3.1 Filter

3.1.1 Introduction

To better prepare the signal for analysis purpose, it is common to perform some filtering techniques to remove noise. In ECG signal processing, the source of noise has a large range from baseline wander and power line interference to muscle movement and poor skin conductivity. Baseline wander is a low-frequency noise which is caused by respiration and body movement. It could be removed by applying a high-pass filter at a cut-off frequency lower than heartbeat. Power line interference is a 60 HZ noise from power source, which could be removed by applying a low-pass filter. Muscle movement and poor skin conductivity noise is rather harder to remove as the noise frequency varies a lot with time and almost impossible to be removed by narrow band filtering. It's spectral content also overlap with the PQRST complex of ECG, which may cover some information in the ECG signal.

3.1.2 Options

To remove baseline wander and power line interference noise, a high-pass filter and a low-pass filter need to be implemented, which can be combined into one band-pass filter. The cut-off frequencies are determined by respiratory rate, heartbeat, and power line frequency. As for the muscle noise and other form of noise overlapping with normal ECG frequency, it is hard to remove without distracting the original ECG pattern, and therefore remained unfiltered in this task of classification. The later steps of CNN classification will extract higher-level patterns and will tolerate some random noise from the dataset.

3.1.3 Design Description

The first step is to use Fourier transform to check the frequency component in a sample signal. FFT (Fast Fourier Transform) is used to generate a Fourier Spectrum, which shows frequency magnitude in dB or 20*log10(magnitude). Fig.3.1 and Fig.3.2 show Fourier Spectrum of the sample normal and AFib signals. The sample signals are divided into three equal length before performing the FFT in order to cancel random noise in the signal. The Fourier Spectrum is then generated with the average of FFT from these three pieces. In Fig.3.1, the heartbeat is measured to be at 1.3 HZ with strong second and third harmonics shown in red box. The blue box shows the low frequency noise that might be caused by respiration or body movement at 0.4 HZ and 0.6 HZ. In Fig.3.2, multiple heartbeat rate is detected with the red box and low frequency noise is again shown in blue box. The peak at around 4 HZ should be a harmonic of the regular heartbeat.

From the Fourier Spectrum, it is clear that ECG signal has less pattern at higher

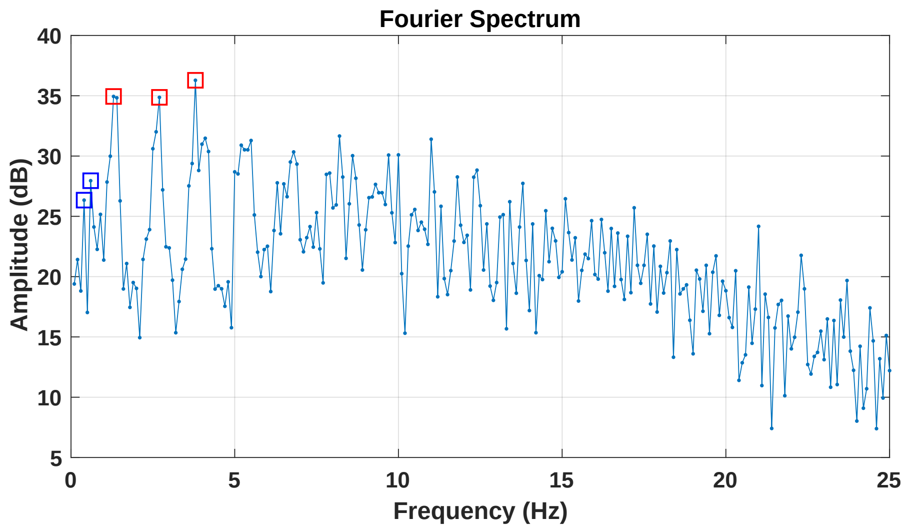

Figure 3.1: Fourier Spectrum of a sample normal signal. The red boxes represent the heartbeats and their harmonics and the blue boxes represent low frequency noise.

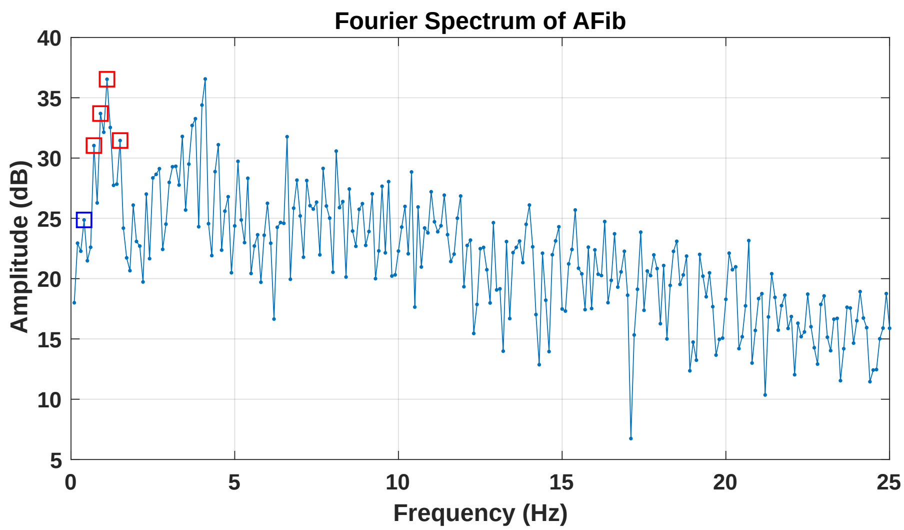

Figure 3.2: Fourier Spectrum of a sample AFib signal. The red boxes represent the heartbeats and the blue boxes represent low frequency noise.

frequency over 10 HZ. A bandpass filter of 0.5 HZ to 10 HZ is then designed to filter these ECG signals. With a built-in function ‘bandpass’ from MATLAB, a bandpass filter with infinite impulse response (IIR), steepness of 0.95, and stopband attenuation at 60 dB is generated to filter all signals from the database.

3.1.4 Analysis and Test Result

The filtered signal from bandpass filter is plotted in Fig.3.3. The low frequency noise from respiration and high frequency noise from power line is eliminated from raw signal, which makes it easier for further analysis and classification.

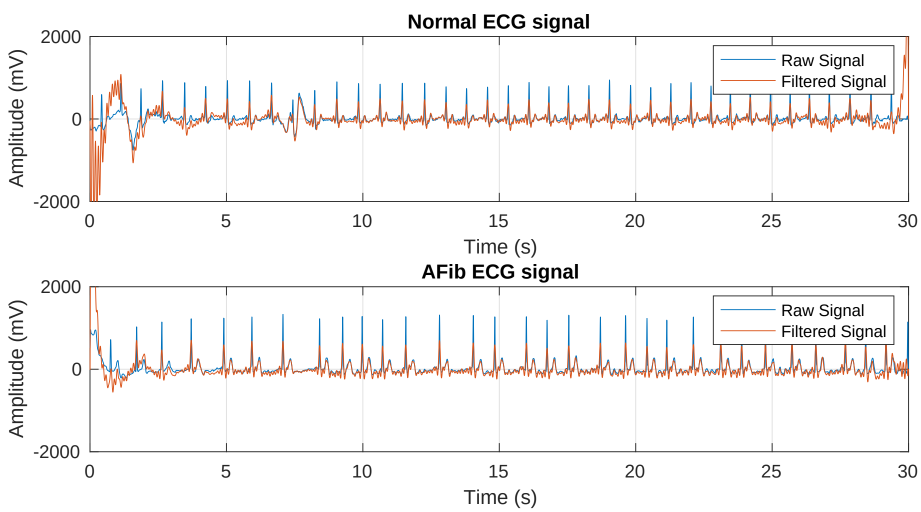

Figure 3.3: Sample signals filtered by bandpass filter.

3.2 Hilbert Diagram

3.2.1 Introduction

Hilbert Transform is a useful tool in signal processing as it can help analyze nonlinear systems and non-stationary signals. It defines instantaneous frequency and amplitude and construct an analytic complex time signal from signal in real time. A phase plot can then be generated from this complex time signal to potentially reveal some pattern in ECG signal. Some previous researches have shown that Hilbert Transform could be used for ECG analysis in QRS complex detection [12] and de-noising [13].

3.2.2 Options

Yan et.al [14] showed previously that a phase plot of Hilbert Transform of the vocal-fold vibration signal could be useful to classify vocal-fold pathology. Similar methodology is applied here to potentially show some pattern of ECG signals. However, the vocal fold signal is much more periodic than ECG signal, and therefore could generate a clearer phase plot. The parameters need to be fine tuned to show a pattern in the ECG phase plot. Hilbert Transform is a linear operator given by convolution with function 1/(πt):

$$H(u)(t) = \frac{1}{\pi} * \int_{-\infty}^{\infty} \frac{u(\tau)}{(t-\tau)} d(\tau) \tag{3.1}$$

which convert a real time signal u(t) into a complex time signal H(u)(t).

3.2.3 Design Description

In MATLAB, the Hilbert transform in completed by a function called 'hilbert', which approximated the analytic signal by calculating the FFT of the original signal, replacing coefficients of negative frequencies to zero, and performing inverse FFT. The complex time signal generated from Hilbert Transform is then normalized between -1 and 1 and a phase plot with real part of the signal on x axis and imaginary part of the signal on y axis is then generated to show the some patterns.

3.2.4 Analysis and Test Result

Fig.3.4 shows sample phase plot from Hilbert Transformation of normal and AFib ECG signals. For a normal ECG, a dominant frequency of heartbeat or R-R interval should be detected and a thicker line composed by high density of dots should be visible on phase plot. On the other hand, an AFib ECG signal has inconstant heartbeat and R-R interval, which results a phase plot with dots spread more uniformly within the square. These patterns could be potentially recognized by a CNN in later steps.

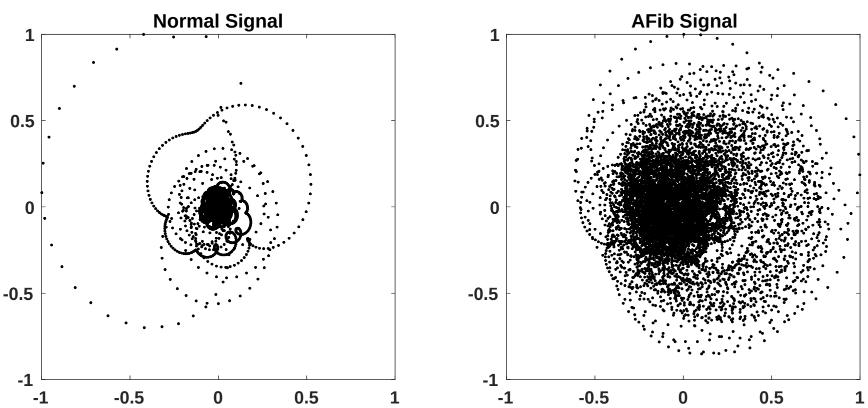

Figure 3.4: A phase plot of normal (left) and AFib (right) ECG signals

3.3 Spectrogram

3.3.1 Introduction

A Spectrogram is a visual representation of a signal where the spectrum of frequencies varies with time. It could show both temporal and frequency information in a same plot with some compromise for both information. A spectrogram is commonly displayed in an image with one axis representing time and the other representing the frequency. The intensity fo signal for any frequency component at any time instant is pseudo-color-mapped. It is widely used in signal processing for its ability to present both temporal and frequency information at same time.

3.3.2 Options

In MATLAB, a spectrogram is generated with short-time Fourier transform. It allows users to change some parameters. The window size is adjusted to divide the signal into segments of certain length, and larger window size compromise more temporal information for more accurate frequency information. The number of overlapped samples sets the overlapping period between each segment. Higher overlapping rate means smoother transition between each data point.

3.3.3 Design Description

For most of signals provided in this dataset, the length is 30 seconds and the sampling frequency is kept at 300 HZ, which means 9,000 data points is collected in one sample. Therefore, I choose to use a 2,000 window size and a 1,900 overlap length between adjoining segments. This setting is optimal to present a clear R-R interval band and its harmonics for normal samples and an oscillating pattern in AFib samples.

3.3.4 Analysis and Test Result

Fig.3.5 shows a sample spectrogram of normal ECG. The strong yellow bands represent constant-rate heartbeats and their harmonics as predicted previously. Fig.3.6 shows a sample spectrogram of AFib ECG. The high intensity lines are random and have no pattern to follow, which is caused by inconstant R-R interval from ECG.

The title, axis names, and color bar will be suppressed for the spectrograms generated for training and testing purpose in later steps.

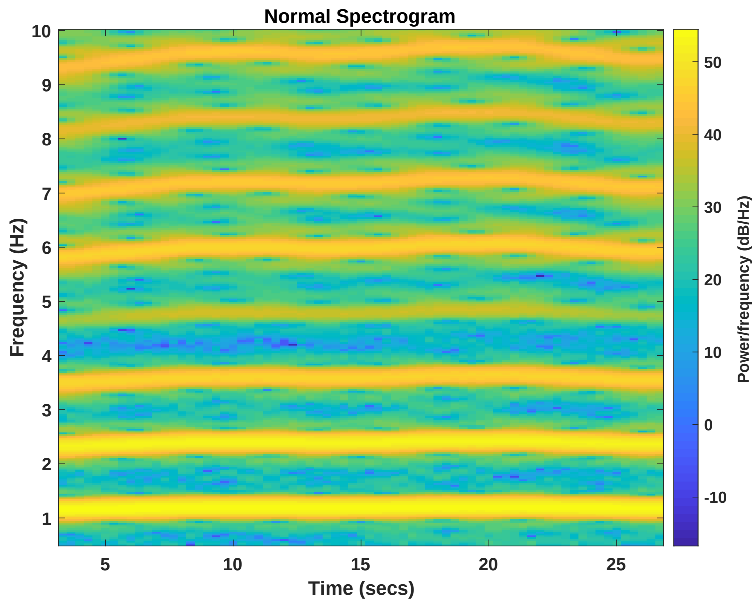

Figure 3.5: A sample spectrogram of normal ECG

3.4 CNN

3.4.1 Introduction

CNN is inspired by biological processes in that the connectivity pattern between neurons and it resembles the organization of the animal visual cortex. It requires less data preprocessing compared to other image classification algorithms. CNN has a large range of applications in image and video recognition, recommend systems, image classification, medical image analysis, and natural language processing (NLP). CNN has advantage in extracting high level features from input using convolution and therefore is an ideal tool to use in this study.

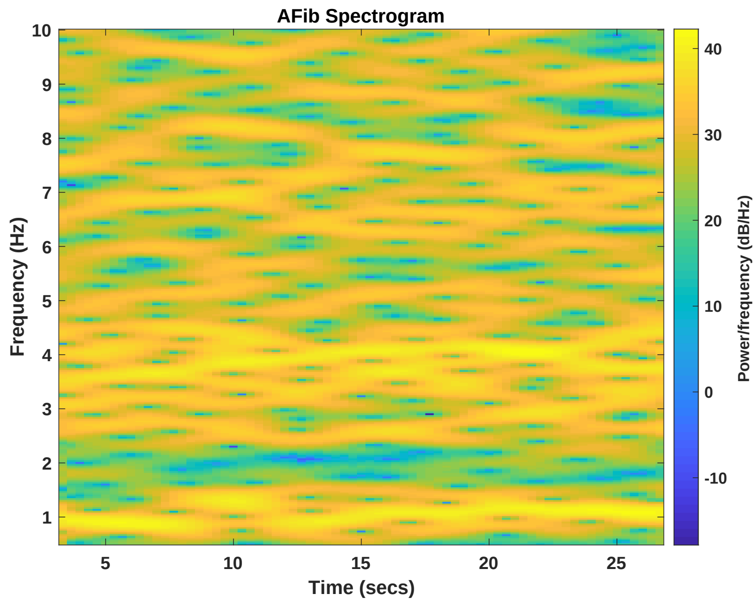

Figure 3.6: A sample spectrogram of AFib ECG

3.4.2 Options

A typical CNN architecture consists of several layers, including convolutional layers, activation layers, normalization layers, pooling layers, and a fully connected layer. The convolutional layers can be altered with different filter size, number of filters, and padding size. The activation layers could be chosen from a ReLU layer, a Sigmoid layer, or a TanH layer. For the normalization layers, a batch normalization or a dropout normalization is commonly used. Max pooling and average pooling are used for pooling layers.

3.4.3 Design Description

For this study, a simple CNN architecture is designed to classify the images generated previously into two categories. A simplified layer graph is shown in Fig.3.7, where conv stands for convolutional layers, BN stands for batch normalization layers, ReLU stands for ReLU layers, MP stands for max pooling layers, and FC stands for fully connected layers.

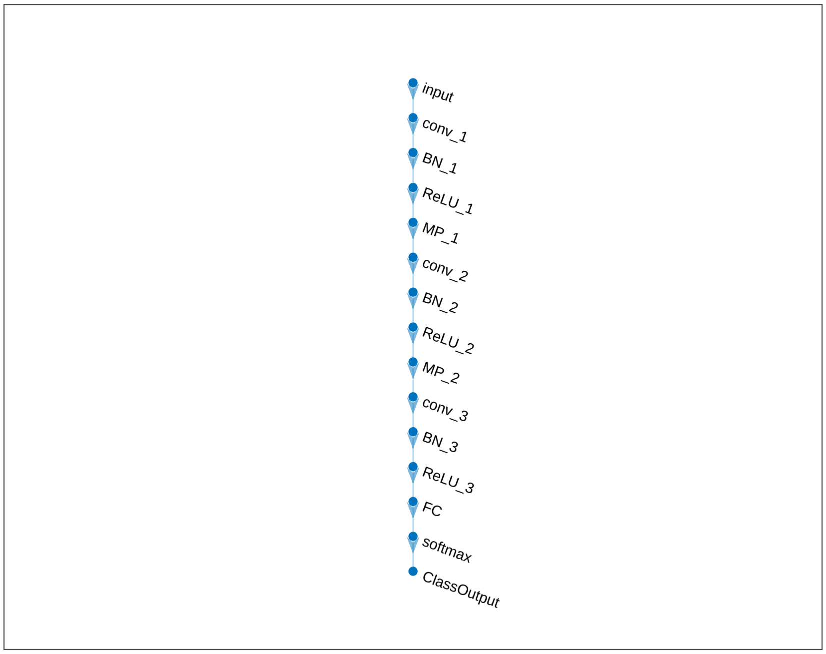

Figure 3.7: A simplified CNN layer graph

3.4.4 Analysis and Test Result

The CNN is applied to classify the Hilbert phase plot and spectrogram of normal and AFib ECG signals. Fig.3.8 shows the training progress of Hilbert phase plot and it reaches above 60% accuracy. Fig.3.9 shows the training progress of spectrogram and it reaches 85.92% accuracy. The spectrogram has better performance than the Hilbert phase plot. Some potential reasons could be that the signals contain too much noise and the CNN is not optimized for this task.

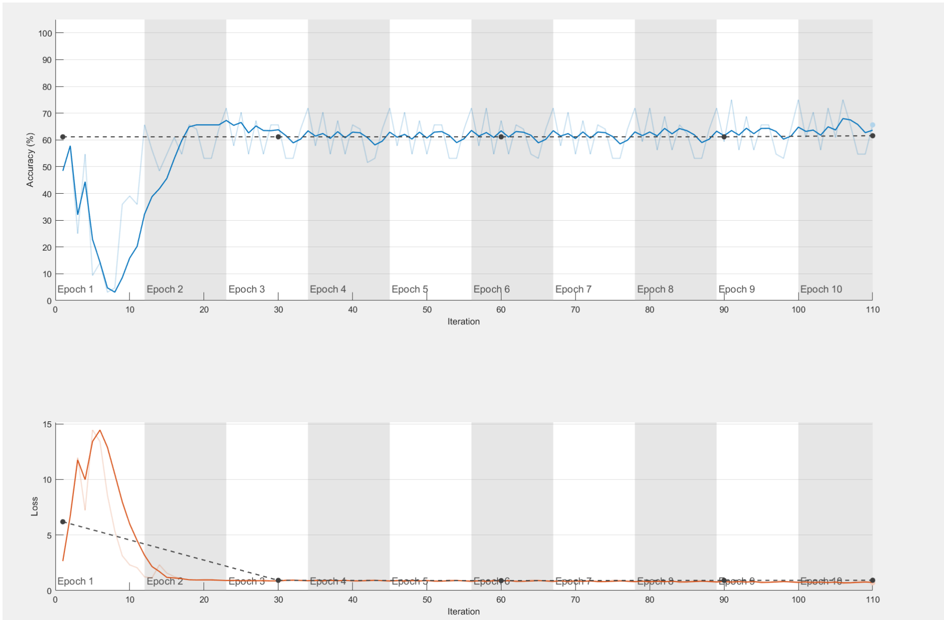

Figure 3.8: The classification result for Hilbert phase plot using a simple CNN

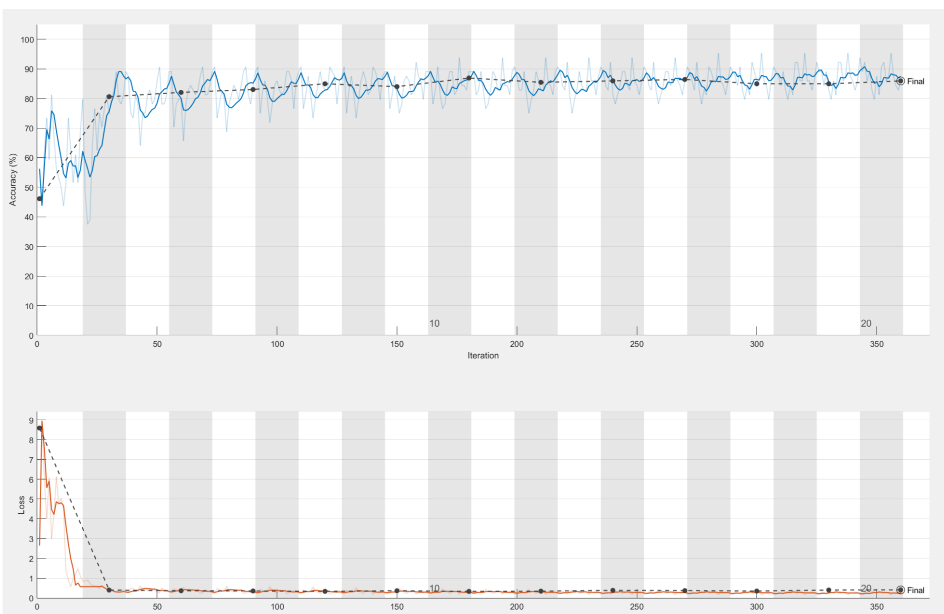

Figure 3.9: The classification result for spectrogram using a simple CNN

3.5 Transfer Learning

3.5.1 Introduction

Transfer learning is a machine learning method where a previously developed model is used as the starting point model for another task. The advantage of this machine learning method is that researchers could use a model that has been fine-tuned by professionals to perform classification tasks on a separate task in their own area. With a limited dataset size, one can hardly train a complex CNN from scratch, and thus using a pretrained model is recommended for a classification task like the one in this study.

3.5.2 Options

There are several famous pretrained CNN models that could be chosen to use in this study. LeNet is one of the earliest successful architectures of CNNs developed by Yann Lecun and was originally used to read digits in images [15]. AlexNet is one the the first popular CNNs in computer vision and was developed by Alex Krizhevsky, Ilya Sutskever and Geoffrey Hinton [16]. GoogLeNet was developed by Christian Szegedy and his team at Google, and has 22 layers with an inception module built by convolutional layers [17]. ResNet was trained on very deep networks (up to 1,200 layers) with all reformulated as learning residual functions with reference to the layer inputs [18].

3.5.3 Design Description

In this study, GoogLeNet Inception 1 was used to classify normal and AFib ECG signals. Fig.3.10 shows a detailed structure of this CNN architecture. All image inputs generated from previous steps, including the Hilbert phase plot and spectrograms, are resized to $224 \times 224 \times 3$, where each image has 224×224 pixels and 3 color layers of red, green, and blue.

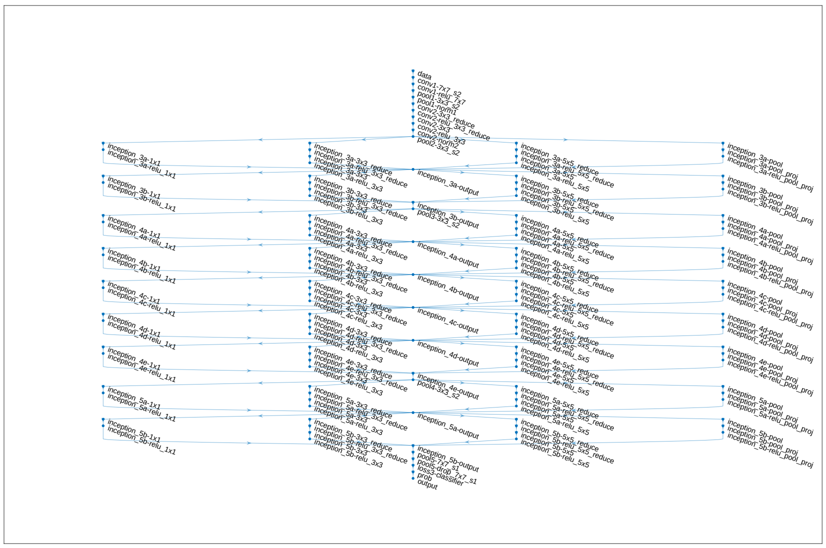

Figure 3.10: A detailed structure of GoogLeNet Inception 1

3.5.4 Analysis and Test Result

This CNN architecture was tested on the spectrogram images. The learning rate was set to be 0.0003 with 20 epochs trained. It took 6 minute and 17 seconds to finish on a single GPU of GeForce GTX 1080 from NVIDIA. A detaied training process is shown in Fig.3.11. It reaches 93.58% accuracy on the independent testing set and 100% accuracy on the training set, which presents an overfitting problem. Fig.3.12 shows some randomly chose testing sample and the predictions made by this network. It also shows the confidence level for making such decisions.

Figure 3.11: The classification result for GoogLeNet Inception 1

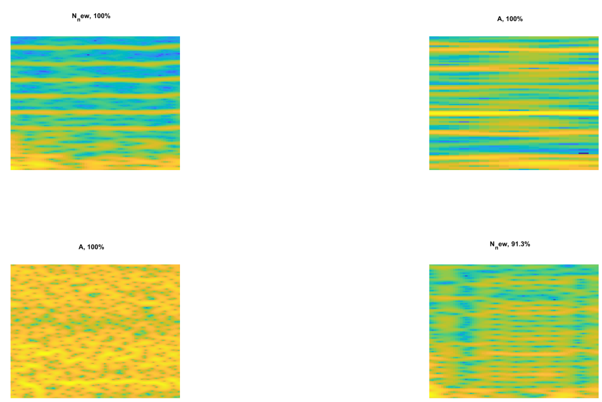

Figure 3.12: A sample classification output for GoogLeNet Inception 1. Nnew stands for normal and A stands for AFib. The percentage next the Nnew or A is the confidence of making the decision

3.6 Data Augmentation

3.6.1 introduction

Data augmentation is a technique used in machine learning to prevent overfitting problem by artificially creating new training data from existing training dataset.

3.6.2 Options

There are several data augmentation techniques available for this study. The image could be reflected among x axis or y axis. It could be also rotated for certain degrees. The image could be scaled uniformly, along x axis, or along y axis. The image could also be sheared or translated horizontally or vertically.

3.6.3 Design Description

In this study, two data augmentation techniques are chosen. The first one is reflection among x axis. It could reflect random noise signal in spectrograms horizontally, which reduces the importance of noisy patterns in the training process. The other techniques used in this study is vertical scaling, which stretches the image on y axis and results in a change of peak frequency from R-R interval. The scaling operation helps reduce the effect of heart rate variation on the training process.

3.6.4 Analysis and Test Result

The augmentation step is added before training the CNN. A random reflection on x axis with 50% probability and a vertical scaling of a factor picked randomly from a continuous uniform distribution of [0.9,1.1] are applied to the training dataset. In other words, the scaling factor is chosen from $1 \pm 10\%$ The learning rate is set to be 0.0003 with 20 epochs trained. It takes 6 minute and 48 seconds to finish on a single GPU of GeForce GTX 1080 from NVIDIA. A detaied training process is shown in Fig.3.13. It reaches 91.89% accuracy on the independent testing set without overfitting problem. Some oscillation is observed in the training progress. Fig.3.14 shows some randomly chose testing sample and the predictions made by this network. It also shows the confidence level for making such decisions.

Figure 3.13: The classification result for GoogLeNet Inception 1 with data augmentation

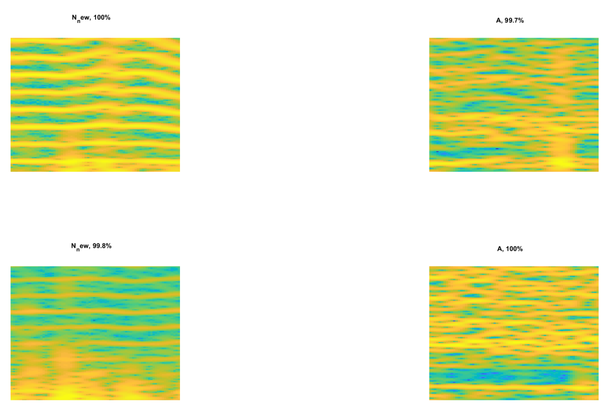

Figure 3.14: A sample classification output for GoogLeNet Inception 1 with data augmentation. Nnew stands for normal and A stands for AFib. The percentage next the Nnew or A is the confidence of making the decision

Chapter 4

Results

4.1 Protocol

From the previous testing in different methods, a protocol of training and testing in AFib detection from ECG signals is created. For this study, all 758 AFib ECG signals and 758 randomly chosen ECG signals from 5,076 normal ECG signals are used to train and test. The testing dataset consists of 20% of all signals and is chosen randomly.

4.1.1 Training

1. The signals from training dataset go through a bandpass IIR filter of 0.5 HZ to 10 HZ with a steepness of 0.95 and stopband attenuation at 60 dB.

2. Using short-time Fourier transform, the signals are represented into spectrograms with window size of 2,000 and overlapping size of 1,900. This setting could be changed base on signal length.

3. The spectrograms are then augmented with vertical reflection and scaling. The scaling factor is set to 1 ± 10%. This step creates more data for training

purpose.

4. All training images are rescaled to $224 \times 224 \times 3$ for training purpose.

5. The training dataset is fed to GoogLeNet Inception 1 to training. The learning rate is set to be 0.0001 and epoch is set to be 40.

4.1.2 Testing

1. The signals from testing dataset go through a bandpass IIR filter of 0.5 HZ to 10 HZ with a steepness of 0.95 and stopband attenuation at 60 dB.

2. Using short-time Fourier transform, the signals are represented into spectrograms with window size of 2,000 and overlapping size of 1,900. This setting should match the setting of training set.

3. All testing images are rescaled to $224 \times 224 \times 3$ for testing purpose.

4. The testing dataset is fed to GoogLeNet Inception 1 to classify.

4.2 Results

The final result of the whole classification algorithm is shown in Fig.4.1. The training environment is using a single GPU NVIDIA GeForce GTX 1080. The total training time is 14 minutes and 12 seconds. The final accuracy on the independent testing dataset is 92.91%, where the testing dataset is randomly chosen from the PhysioNet challenge dataset and consists 20% of it. The receiver operating characteristic (ROC) curve is also generated for this model, which is shown in Fig.4.2. The AUROC (area under ROC curve) is measured to be 0.9789.

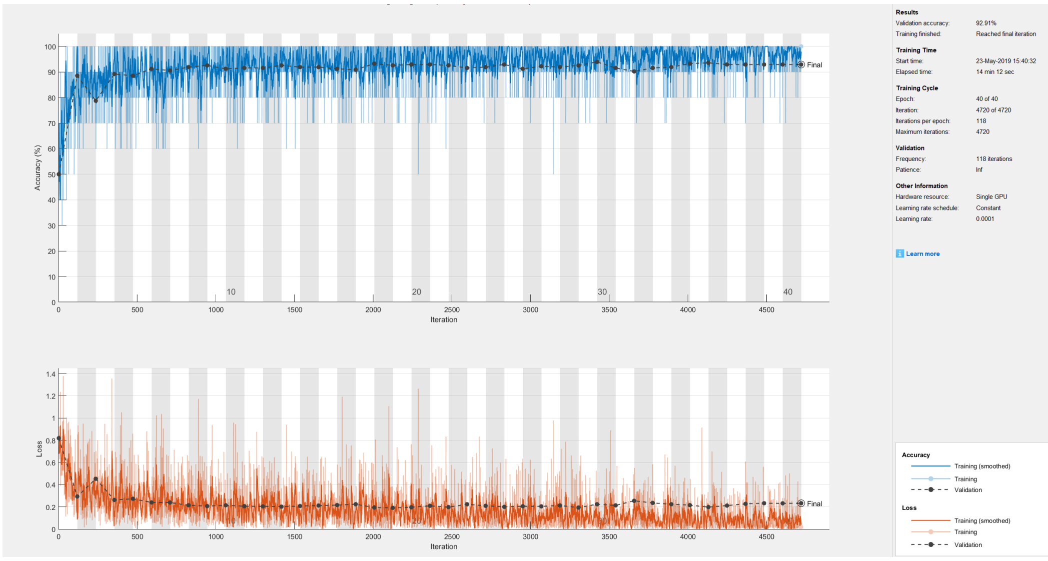

Figure 4.1: The training progress plot for classification

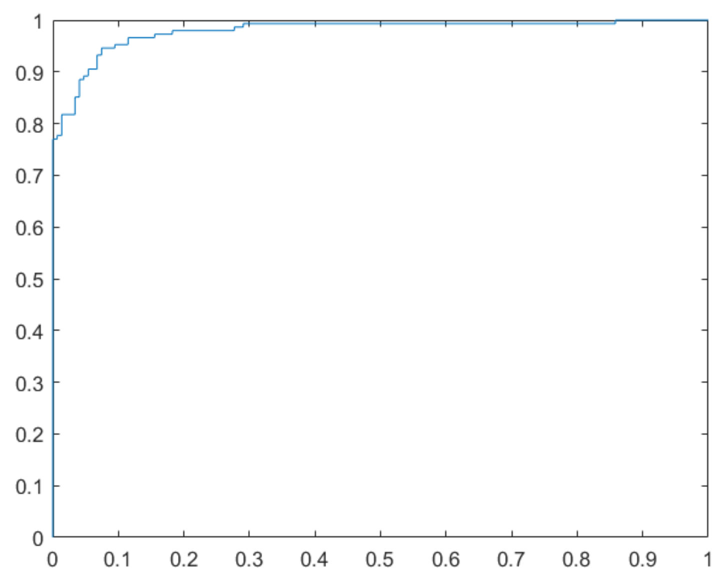

Figure 4.2: The ROC curve for classification

4.3 Discussion

In this study, the training accuracy is still oscillating as shown in Fig.4.1. This could be caused by insufficient training epochs. Some artificial noise and muscle noise from the original dataset has still influenced in the trained model, which lowers the accuracy.

The dataset used in this study contains only 758 AFib ECG signal, which is not a large number for CNN training and image classification tasks. The label is created by a single expert, which may be biased and not accurate. These factors all negatively impact the result and may be improved by further study.

Chapter 5

Conclusion

In this thesis study, a novel algorithm was presented for the detection of AFib using ECG signals, and the algorithm was shown to achieve a high detection accuracy. The methodology used in this study, which utilize image classification methods (e.g. CNN) for signal classification, also has great potential in applying to other areas for computational biology.

There are several future directions this project can head for, which include using a larger dataset to train the CNN model in order to achieve further improved performance. In addition, the parameters of the Hilbert transform-based phase plot and the spectrogram could be varied to generate more data for the CNN training. Finally, a better CNN architecture or other pre-trained models could be used for this classification task to improve either accuracy or/and efficiency.

From this thesis study, I learned a lot of professional knowledge in biology and machine learning, as well as the skills to conduct a research project. With more and more cutting-edge and interdisciplinary research coming in the future, I am sure that this knowledge and skills will become useful and handy.

Bibliography

[1] C. T. January, L. S. Wann, J. S. Alpert, H. Calkins, J. E. Cigarroa, J. C. Cleveland, J. B. Conti, P. T. Ellinor, M. D. Ezekowitz, M. E. Field, *et al.*, "2014 aha/acc/hrs guideline for the management of patients with atrial fibrillation: a report of the american college of cardiology/american heart association task force on practice guidelines and the heart rhythm society," *Journal of the American College of Cardiology*, vol. 64, no. 21, pp. e1–e76, 2014.

[2] W. B. Kannel, P. A. Wolf, E. J. Benjamin, and D. Levy, "Prevalence, incidence, prognosis, and predisposing conditions for atrial fibrillation: population-based estimates," *The American journal of cardiology*, vol. 82, no. 7, pp. 2N–9N, 1998.

[3] S. Stewart, C. L. Hart, D. J. Hole, and J. J. McMurray, "A population-based study of the long-term risks associated with atrial fibrillation: 20-year follow-up of the renfrew/paisley study," *The American journal of medicine*, vol. 113, no. 5, pp. 359–364, 2002.

[4] J. Gajewski and R. B. Singer, "Mortality in an insured population with atrial fibrillation," *Jama*, vol. 245, no. 15, pp. 1540–1544, 1981.

[5] P. A. Wolf, R. D. Abbott, and W. B. Kannel, "Atrial fibrillation as an independent risk factor for stroke: the framingham study.," *Stroke*, vol. 22, no. 8, pp. 983–988, 1991.

[6] E. J. Benjamin, S. S. Virani, C. W. Callaway, A. M. Chamberlain, A. R. Chang, S. Cheng, S. E. Chiuve, M. Cushman, F. N. Delling, R. Deo, *et al.*, "Heart disease and stroke statistics-2018 update: a report from the american heart association.," *Circulation*, vol. 137, no. 12, p. e67, 2018.

[7] J. S. Healey, S. J. Connolly, M. R. Gold, C. W. Israel, I. C. Van Gelder, A. Capucci, C. Lau, E. Fain, S. Yang, C. Bailleul, *et al.*, "Subclinical atrial fibrillation and the risk of stroke," *New England Journal of Medicine*, vol. 366, no. 2, pp. 120–129, 2012.

[8] W. J. Verberk, S. Omboni, A. Kollias, and G. S. Stergiou, "Screening for atrial fibrillation with automated blood pressure measurement: research evidence and practice recommendations," *International journal of cardiology*, vol. 203, pp. 465–473, 2016.

[9] F. Rincón, P. R. Grassi, N. Khaled, D. Atienza, and D. Sciuto, "Automated real-time atrial fibrillation detection on a wearable wireless sensor platform," in *2012 Annual International Conference of the IEEE Engineering in Medicine and Biology Society*, pp. 2472–2475, IEEE, 2012.

[10] S. Hong, M. Wu, Y. Zhou, Q. Wang, J. Shang, H. Li, and J. Xie, "Encase: An ensemble classifier for ecg classification using expert features and deep neural networks," in *2017 Computing in Cardiology (CinC)*, pp. 1–4, IEEE, 2017.

[11] G. D. Clifford, C. Liu, B. Moody, H. L. Li-wei, I. Silva, Q. Li, A. Johnson, and R. G. Mark, "Af classification from a short single lead ecg recording: the physionet/computing in cardiology challenge 2017," in *2017 Computing in Cardiology (CinC)*, pp. 1–4, IEEE, 2017.

[12] D. Benitez, P. Gaydecki, A. Zaidi, and A. Fitzpatrick, "The use of the hilbert transform in ecg signal analysis," *Computers in biology and medicine*, vol. 31, no. 5, pp. 399–406, 2001.

[13] J. Tang, Q. Zou, Y. Tang, B. Liu, and X.-k. Zhang, "Hilbert-huang transform for ecg de-noising," in *2007 1st International Conference on Bioinformatics and Biomedical Engineering*, pp. 664–667, IEEE, 2007.

[14] Y. Yan, K. Ahmad, M. Kunduk, and D. Bless, "Analysis of vocal-fold vibrations from high-speed laryngeal images using a hilbert transform-based methodology," *Journal of voice*, vol. 19, no. 2, pp. 161–175, 2005.

[15] Y. LeCun, B. E. Boser, J. S. Denker, D. Henderson, R. E. Howard, W. E. Hubbard, and L. D. Jackel, "Handwritten digit recognition with a back-propagation network," in *Advances in neural information processing systems*, pp. 396–404, 1990.

[16] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Advances in neural information processing systems*, pp. 1097–1105, 2012.

[17] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich, “Going deeper with convolutions,” in *Computer Vision and Pattern Recognition (CVPR)*, 2015.

[18] K. He, X. Zhang, S. Ren, and J. Sun, “Deep residual learning for image recognition,” in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 770–778, 2016.