Alexandria University- Faculty of Engineering

Computer and Communication Engineering Department

Heartbeat Sounds Assignment Report

Pattern Recognition Course

Eyad Salama 7128

Nour Hesham 7150

Abstract

Both pattern recognition and artificial intelligence benefit greatly from audio categorization, both theoretically and practically. We present a unique audio categorization approach based on machine learning techniques in this research. First, we show the hierarchical structure of audio data, which consists of four layers: 1) Audio frame, 2) Audio clip, 3) Audio shot, and 4) Audio semantic unit at the highest level. Second, three types of audio data features are recovered in order to build a feature vector: 1) short time energy, 2) zero crossing rate, and 3) Mel-Frequency cepstral coefficients.

Analysing audio recordings is what's meant to be done throughout the process of audio classification, which is also known as sound classification. This incredible method may be used for a variety of purposes in the fields of artificial intelligence and data science, including the creation of chatbots, automatic voice translators, virtual assistants, the classification of musical genres, and apps that convert text to speech.

Classifications of sound can come in a variety of shapes and sizes, including the following: acoustic data classification, also known as acoustic event detection; music classification; natural language classification; environmental sound classification; and environmental sound classification.

Table of Contents

Table of Figures

Chapter 1: Introduction

1.1 Audio Files Development

People now have access to a vast amount of multimedia information because of the Internet, which has also led to the creation of an extremely large-scale multimedia information database.

Because it is so difficult to construct a description of multimedia content and retrieve it afterwards, there is a requirement for some type of efficient retrieval mechanism that is tailored to multimedia. The most important question in the field of multimedia information retrieval is how to assist individuals most efficiently in locating the desired multimedia information in a manner that is both speedy and accurate. In this context, a content-based information retrieval system has been developed; this system may be thought of as a sort of emerging retrieval technology.

The term "audio information" refers to one of the most significant channels via which humans take in information. Due to the exponential growth of audio and video data, it is of the utmost importance to immediately categorize large numbers of audio recordings according to their semantic descriptions.

The recovery of audio is far more challenging than the retrieval of images and videos due to the intricacy of the format. Because the original audio data is a non-semantic and non-structured binary stream, it does not have a semantic description, nor is it organized in a structured fashion.

In addition, audio data is characterized by having a complicated structure, a large amount of data, and a significant necessity for data processing.

As a consequence of this, it is extremely challenging to do in-depth processing and analysis on audio data, and it is also challenging to develop systems that perform audio retrieval and content filtering. The extraction of structured information and semantic elements from audio data is of utmost relevance.

1.2 Description of Audio Data

In this section, we will be analyzing the primary characteristics of audio data as well as give other domains that we typically use while processing through audio data.

Additionally, the hierarchical structure of audio data includes

1. Audio high level semantic unit.
2. Audio shot.
3. Audio clip.
4. Audio frame.

Audio frame: Audio is a non-stationary random process, and the qualities it has fluctuate as a function of the passage of time. On the other hand, its rate of change is relatively sluggish. As a consequence of this, the processing of the audio signal can be broken up into a number of shorter intervals. The term "audio frame" refers to these brief intervals, which are typically between 20 and 30 milliseconds in length. The audio frame is the smallest unit used in the audio processing process.

A new audio unit with a bigger time granularity should be defined since the time granularity of the audio frames is too little and it is impossible to extract significant semantic elements from them. (Audio clip) (named as **Audio clip**).

A number of frames, each of which has a predetermined amount of time allotted to it, compose an audio clip. On the basis of the audio frame, the properties of the audio segment may be computed.

Audio cut: This idea was conceived based on the content of the video clip. Due to the fact that individual audio segments are too brief, this material cannot be used for semantic content analysis. An audio shot is a structural unit of sound that has the same audio class throughout its whole.

The following is an audio high level semantic unit: It is a structural unit of audio that has rich semantic contents and is generated by the various distinct combinations of the audio clips.

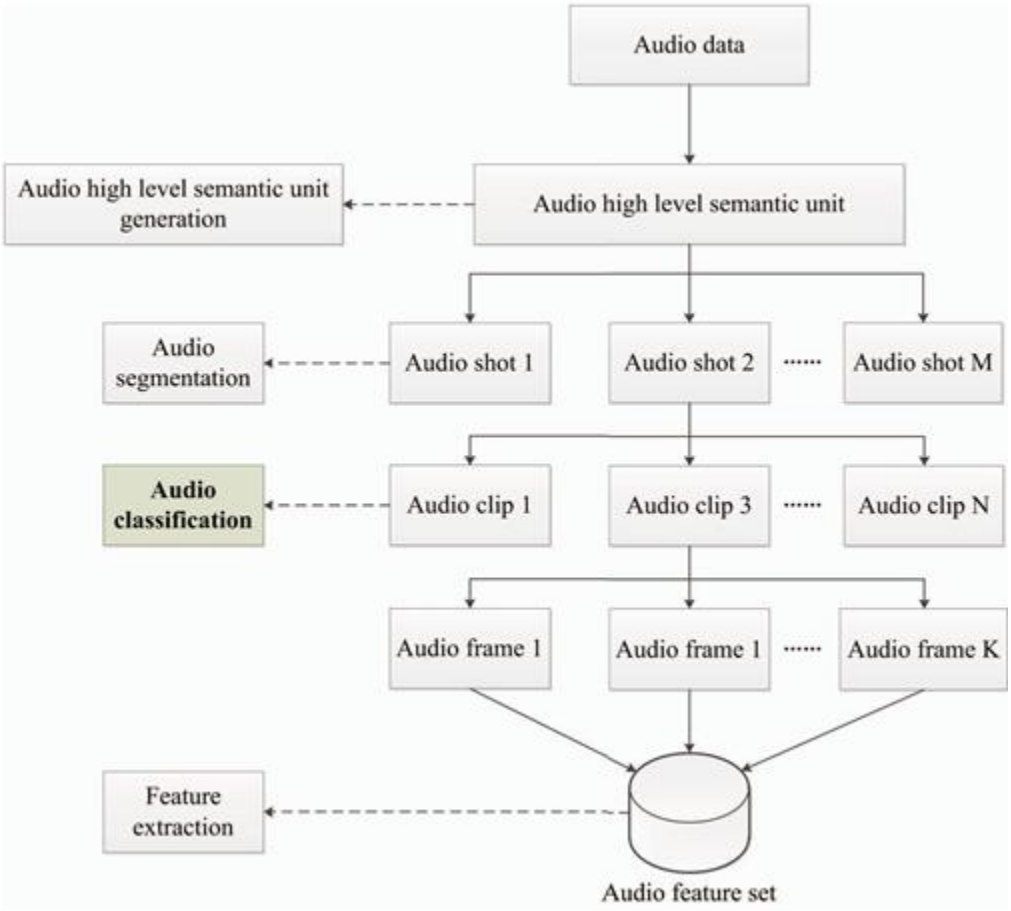

Figure 1: Hierarchical Structure of Audio Data

1.2.1 Spectrogram

A spectrogram is a visual representation of the signal intensity, sometimes known as the "loudness," of a signal across time at the various frequencies that are present in a specific waveform. This representation is known as a spectral analysis. Not only is it possible to determine if there is more or less energy at a certain frequency, such as 2 Hz vs 10 Hz, but it is also possible to determine how the amounts of energy change over the course of time. In fields of study other than physics, spectrograms are frequently utilised to depict the frequencies of sound waves captured by microphones from sources such as humans, machines, animals, whales, and aircraft, amongst other things. In the field of seismology, spectrograms are increasingly being used to analyse the frequency content of continuous signals recorded by individual seismometers or groups of seismometers. This can assist in distinguishing between and characterising the various types of earthquakes and other types of earth vibrations.

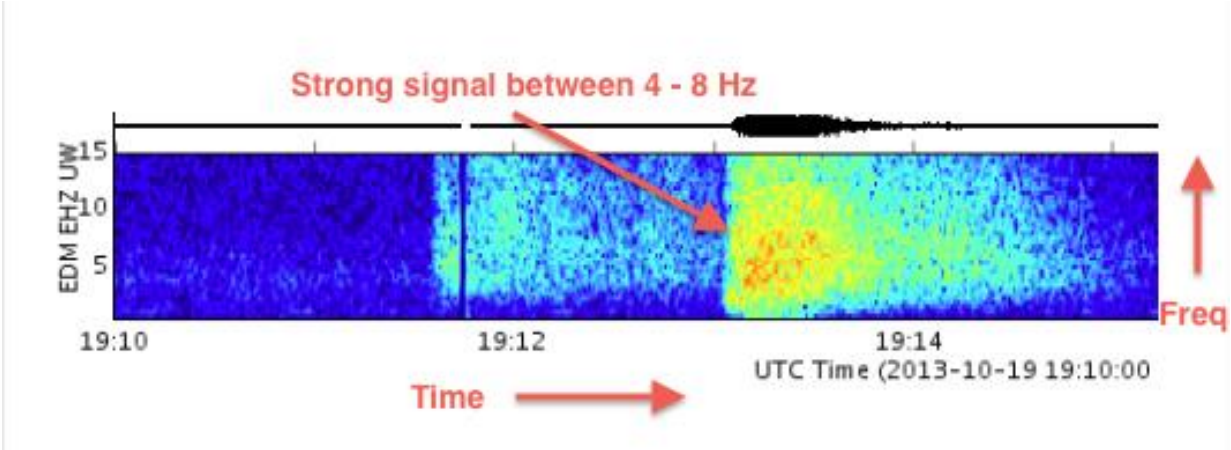

Figure 2: Spectrogram Preview

1.2.2 Mel Frequency Cepstral Coefficient

Having never worked in the field of speech processing, hearing the term "MFCC" (which is frequently used by peers) left me with an incomplete idea that it is the name given to a certain type of "feature" derived from audio data (similar to edges that constitute a kind of feature extracted from images).

Taking the log of the magnitude of this Fourier spectrum and then transforming it using a cosine transformation

Wherever there is a periodic element in the original time signal, we see a peak. Because we apply a transform to the frequency spectrum, the resultant spectrum is neither in the frequency domain nor in the time domain, hence Bogert et al. named it the quefrency domain. Cepstrum is the name given to the spectrum of the log of the spectrum of the time signal.

Cepstrum was initially used to analyze seismic echoes caused by earthquakes.

The Mel scale is a scale that compares a tone's perceived frequency to its actual measured frequency. It adjusts the frequency to better fit what the human ear can hear (humans are better at identifying small changes in speech at lower frequencies).

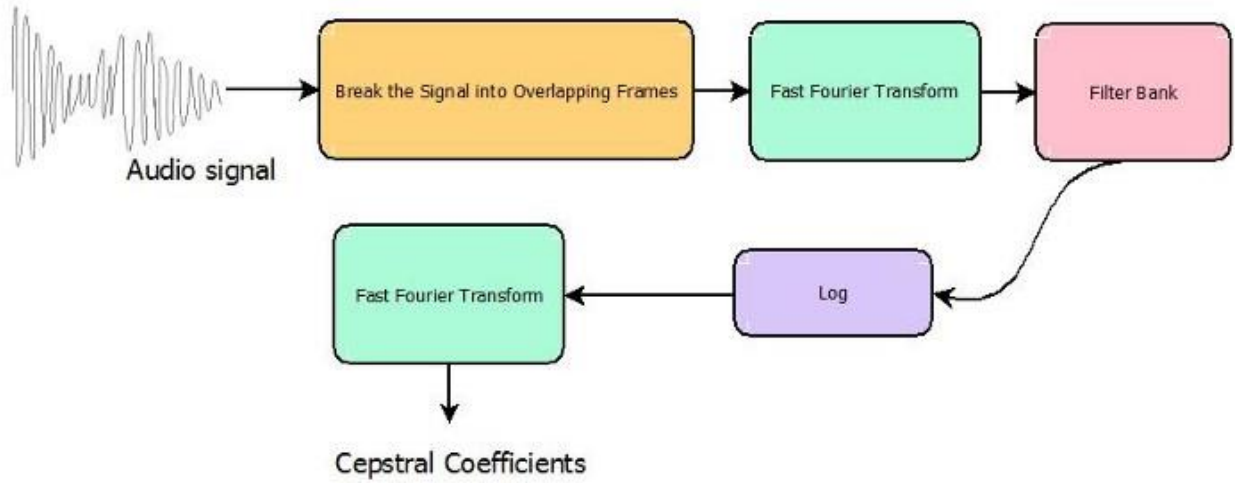

Figure 3: Cepstral Coefficients

1.3 Electrocardiogram

Electrocardiogram, often known as an EKG or ECG, signals are waveforms that depict the electrical activity of the human heart (P, QRS, and T).

The duration, shape, and distances between distinct peaks of each waveform are utilized to identify cardiac disease.

To improve the analysis of ECG data, a novel technique based on two-event related moving averages (TERMA) and fractional Fourier transform (FrFT) algorithms is suggested in this paper. The TERMA algorithm defines specific regions of interest in order to find the required peak, whereas the FrFT spins ECG data in the time-frequency plane to reveal the locations of numerous peaks. The suggested approach outperforms state-of-the-art techniques in terms of performance. Furthermore, estimated peaks, durations between various peaks, and other ECG signal parameters were utilized to develop a machine-learning model to automatically identify cardiac illness. The MIT-BIH database is used in the majority of the accessible research (only 48 patients).

However, in this study, the newly disclosed Shaoxing People's Hospital (SPH) database was utilized to train the suggested machine-learning model, which is more realistic for categorization. The novelty of our proposed machine-learning approach is the cross-database training and testing with promising outcomes.

Figure 4: EKG of Normal Heartbeat

The first wave, known as a "P wave," is produced by the right and left atria, or top chambers "When an electrical impulse travels to the bottom chambers or ventricles, it follows a flat line.

The following wave is a "QRS complex," which is formed by the right and left bottom chambers." The last wave, or "T wave," signifies the ventricles' electrical recovery or return to a resting condition.

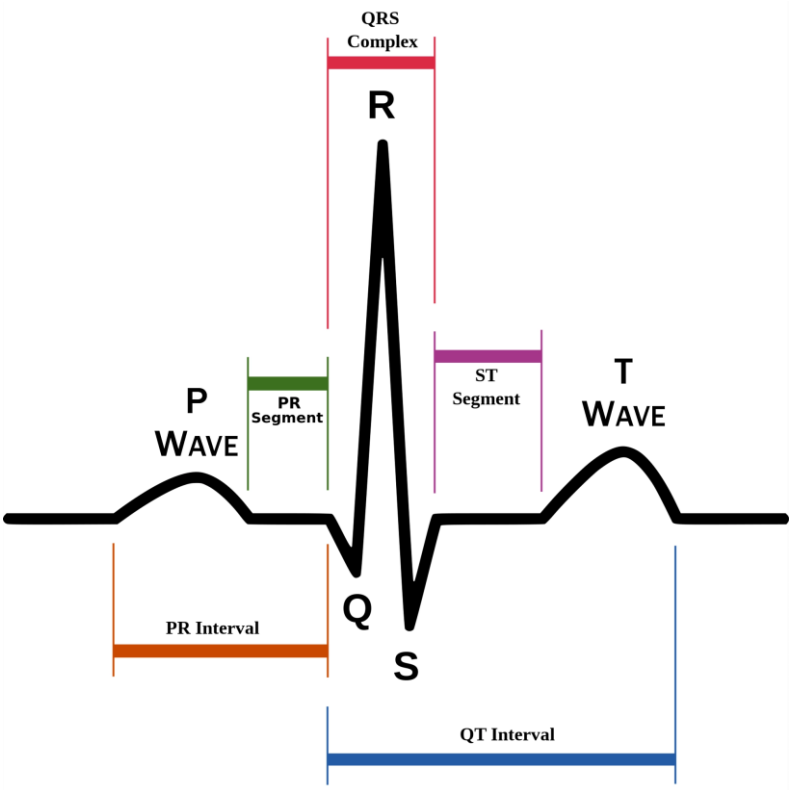

Figure 5: EKG Signal Details

1.3.1 Motivation behind Electrocardiogram

An EKG gives two types of information. First, a health care provider can calculate how long the electrical wave takes to move through the heart by monitoring time intervals on the EKG. The length of time it takes a wave to go from one portion of the heart to the next indicates if the electrical activity is normal, slow, rapid, or irregular. Second, a cardiologist may be able to determine if sections of the heart are damaged, overly big, or overworked by evaluating the amount of electrical activity travelling through the heart muscle.

1.4 Electrocardiogram Analysis Methods

1.4.1 Fast Fourier Transform

Mathematically, the process of Fourier analysis is represented by the Fourier transform

$$\int_{-\infty}^{\infty} F(k)\, e^{2\pi i k x}\, dk$$

Figure 6: Fourier Transform General Rule

The FFT method uses normally DFT method to satisfy the procedure to plot the signal in frequency domain both in frequency response and phase response.

The L-point DFT is sufficient to uniquely represent the sequence of the discrete signal in the frequency domain, it is apparent that it does not provide sufficient detail to yield a good picture of the spectral characteristics of the signal. If we wish to have better picture, we must interpolate the frequency response at more closely spaced frequencies. In fact, we can view this computation as expanding the size of the sequence L points to N points by appending N-L zeros to the discrete sequence, that is, zero padding.

Then the N-point DFT provides finer interpolation than the L-point DFT. The continuous time domain ECG signal has been distorted to discrete time domain signal using sampling theorem. The discrete values and the DFT point have been taken to be used in the DFT process. Here the discrete values n are taken in terms the DFT point of m which are belonging to the above two DFT equation. The parameters p and q hold the DFT coefficients that can help to draw the plot of frequency response and the exponential part which carries the phase that can help to draw the plot of the phase response of the ECG signal

1.4.2 Short Time Fourier Transform

To address this issue of FFT, Dennis Gabor initially presented the windowed-Fourier transform, sometimes known as the Gabor transform in 1946. STFT contains information on both time and frequency.

It is used to identify the signal's sinusoidal frequency and phase content as it evolves over time. In compared to other time-frequency analysis techniques, the STFT-based spectrogram is a simple and rapid approach. It is a simple method of dividing the waveform of interest into a number of small pieces. Then it applies the conventional Fourier transform to each segment.

A window function is applied to a data segment, thereby separating it from the overall waveform, then the Fourier transform is applied to that segment. This is referred to as a spectrogram or Short-Time Fourier Transform.

1.4.3 Wavelet Transform

Because the STFT window should always have a fixed size, it does not provide multi resolution information on the signal. However, the Wavelet Transform has a multiresolution characteristic that provides both time and frequency information via changing window size.

Jean Morlet, a French geophysicist, proposed the notion of a 'Wavelet' in 1982. Wavelet refers to a little wave, and the Wavelet Transform is a novel technique for seismic data processing. Alex Grossmann theoretical physicists examined the inverse formula for the wavelet transform right away.

A Wavelet is a tiny wave with concentrated energy in time that may be used to analyze transient, nonstationary, or time-varying signals.

There are several Wavelets available for usage in a wide range of applications. Biorthogonal, Haar, Coiflet, Symlet, Daubechies Wavelets, and other wavelet families are examples. Some of the aspects that make them helpful are:

1. Wavelets are localized in both time and frequency.
2. For analyzing non-stationary signals such as ECG which have frequent level variations and uneven features.
3. Wavelet separates a signal into multiresolution components.

1.5 About Our Dataset

1.5.1 Context

This dataset was originally for a Machine/Deep learning challenge to classify heartbeat sounds. The data was gathered from two sources: (A), and (B).

1. From the general public via the iStethoscope Pro iPhone app.
2. From a clinical trial in hospitals using the digital stethoscope DigiScope.

1.5.2 Challenged Introduced

1. **Heart Sound Segmentation**: The first challenge is to produce a method that can locate S1(lub) and S2(dub) sounds within audio data, segmenting the Normal audio files in both datasets.
2. **Heart Sound Classification**: The task is to produce a method that can classify real heart audio (also known as “beat classification”) into one of four categories.

1.5.3 Content

The dataset is split into two sources, **A** and **B**:

1. **set_a.csv** - Labels and metadata for heart beats collected from the general public via an iPhone app
2. **set*a*timing.csv** - contains gold-standard timing information for the "normal" recordings from Set A.
3. **set_b.csv** - Labels and metadata for heart beats collected from a clinical trial in hospitals using a digital stethoscope
4. **audio files** - Varying lengths, between 1 second and 30 seconds. (some have been clipped to reduce excessive noise and provide the salient fragment of the sound).

Chapter 2: Assignment Progress

2.1 Data Exploratory Analysis

2.1.1 Knowing our Dataset Labels

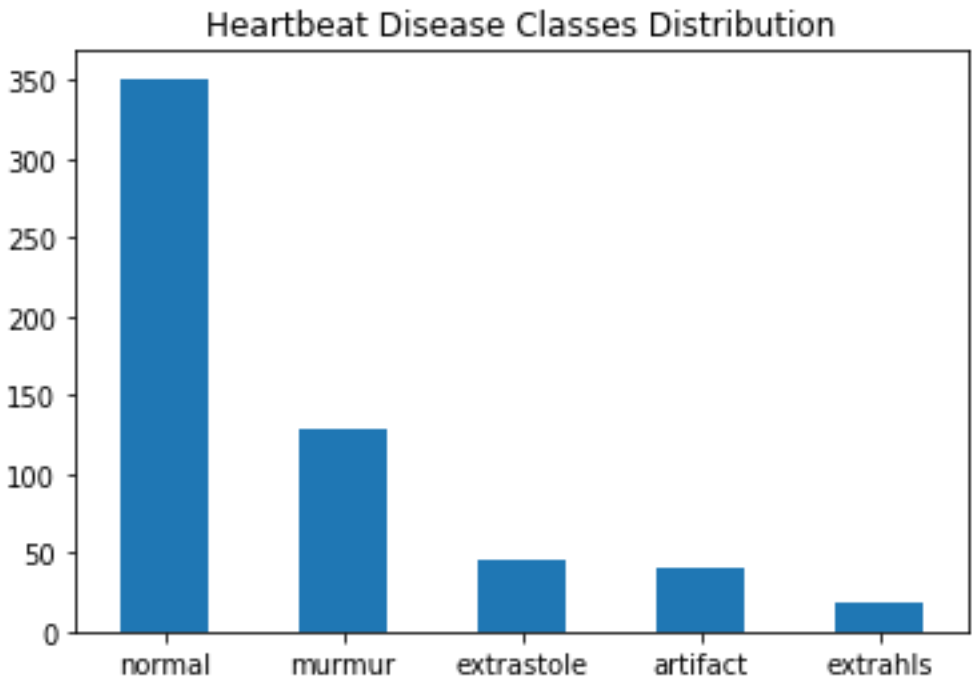

Figure 7: Heartbeat Diseases Class Distribution

2.1.2 Signal Denoising

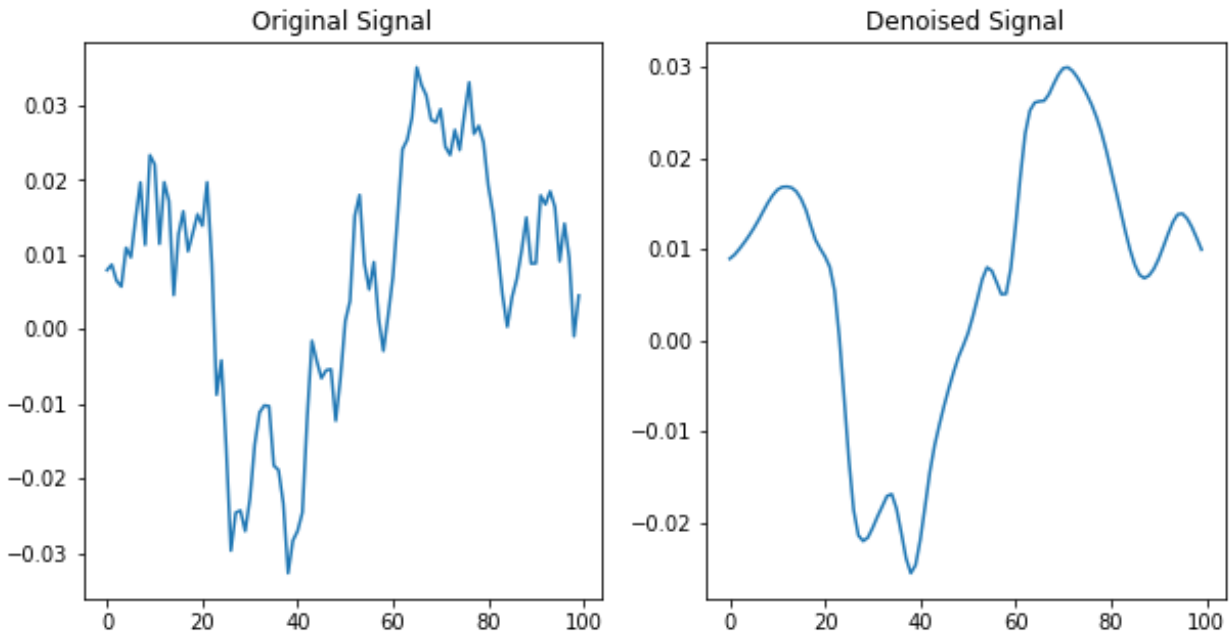

Figure 8: Signal Before and After Denoise

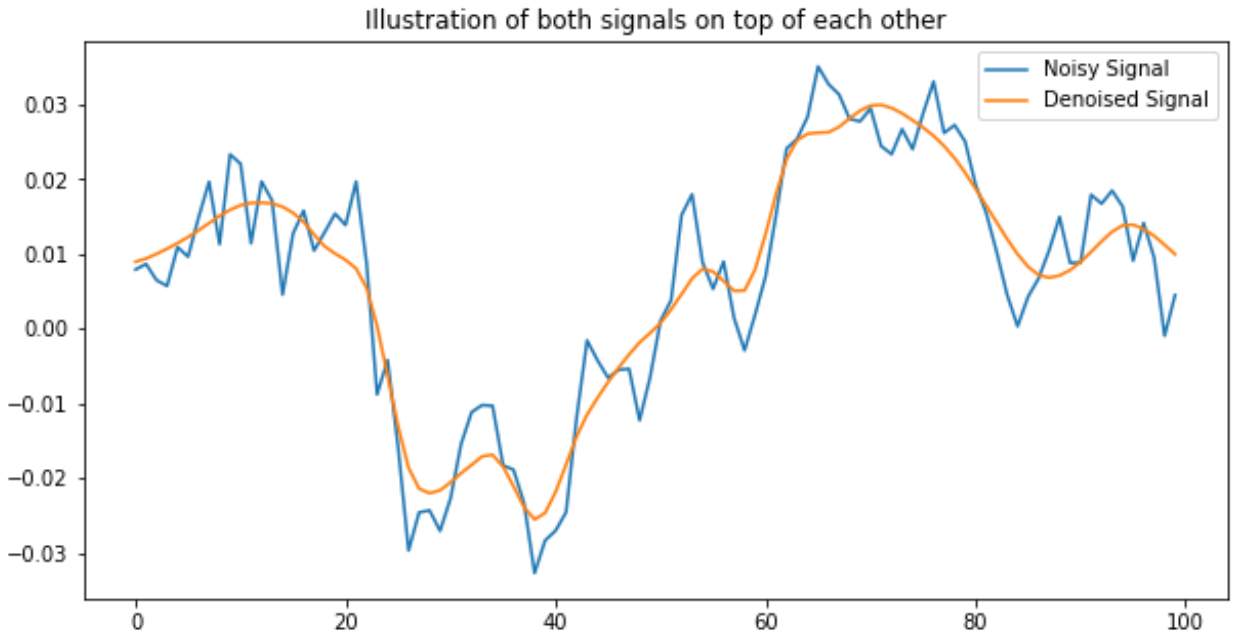

Figure 9: Signals Stacked

2.2 Segmentation

2.2.1 Data Splitting and Reorganization

We read the timing csv file and parse the information in it. We find that we have 21 independent audio samples where each sample is made up from a number of lub-dub cycles.

We divide each audio into a number of audios equal to the number of cycles present in it. We also split the data into three sets: training, validation, and testing.

2.2.2 Dataset Creation

First, we create our Dataset class.

1. Both input and output should be wav files.
2. We added a Mel spectrogram and an MFCC to our dataset if we want our model to handle 2D data instead of 1D data.

Steps:

1. We extract audio features as a spectrogram (time as x-axis, frequency as y axis). A spectrogram shows frequencies in linear scale.
2. We convert to Mel Spectrogram which shows frequencies in Mel scale, since humans discriminate lower frequencies better than higher frequencies. (This is done using Mel filters).
3. We transform amplitude into the decibel scale since humans perceive loudness on a logarithmic scale.

During loading of data:

1. We normalize the data so that the sample values are between -1.0 and +1.0.
2. We convert stereo (two channels) to mono (on channel)
3. We normalize label values to be between 0 and 1 depending on clip length.

2.2.3 OurSegModel

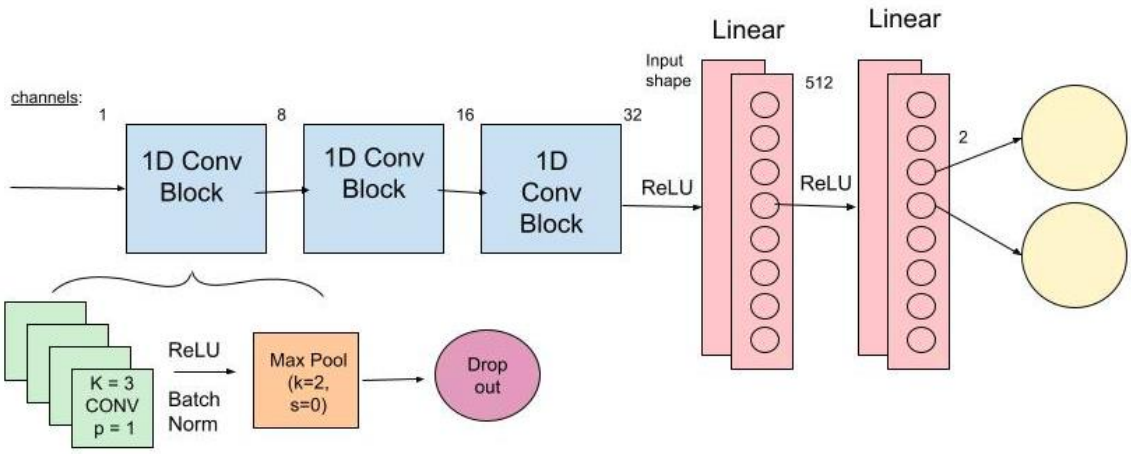

Figure 10: OurSegModel Diagram

2.2.4 OurSegModel2

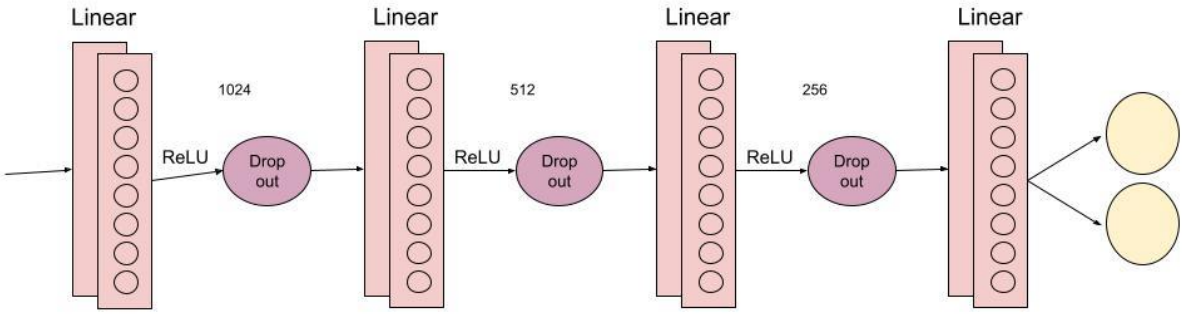

Figure 11: OurSegModel2 Diagram

2.2.5 Segmentation Experiments Best Results

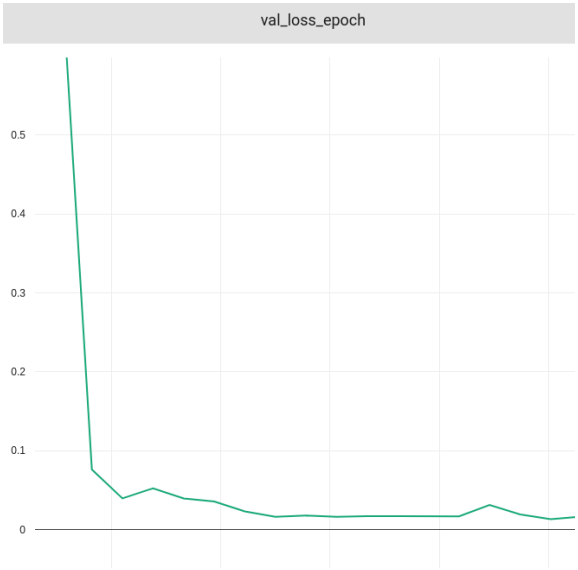

Figure 12: Segmentation Validation Loss

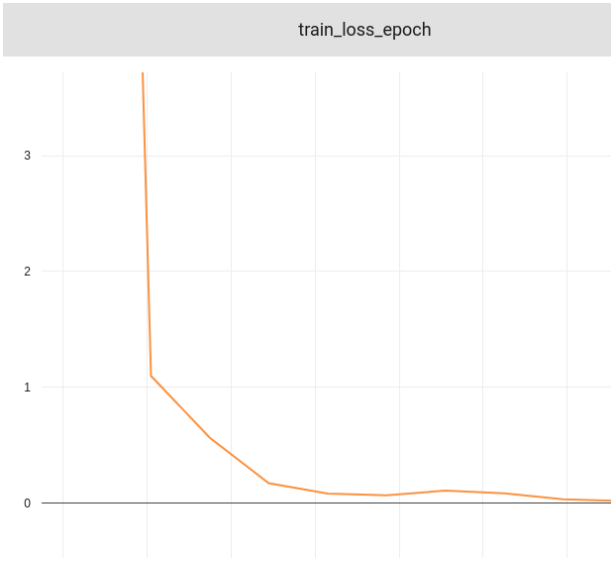

Figure 13: Training Loss

2.2.6 Results and Experiments

ID	Model	denoised	LR	Batch size	Epochs	Data Type	Train MSE (min)	Val MSE (min) *	Val MSE	Test MSE (min)
1	OurSegModel	F	5.00E-03	10	20	WAV	0.009 (s) / 0.02 (e)	0.0028 (s) / 0.013 (e)	0.0167	0.031899
2	OurSegModel	F	1.00E-03	12	20	WAV	0.003 (s) / 0.01 (e)	0.0008 (s) / 0.015 (e)	0.0206	0.038
3	OurSegModel_2	F	5.00E-03	12	50	WAV	0.003 (s) / 0.01 (e)	0.0009 (s) / 0.01004 (e)	0.0167	0.0256
4	OurSegModel_2	F	1.00E-03	8	20	WAV	0.003 (s) / 0.019 (e)	0.001 (s) / 0.013 (e)	0.018	0.038

2.3 Classification

2.3.1 Data Splitting and Reorganization

For the classification task, it was important that our 3 splits were balanced, which means that the general distributions of our sets were similar.

This prevents a class from being present only in the testing set, or only in the training set, which reduces variance (reduces overfitting).

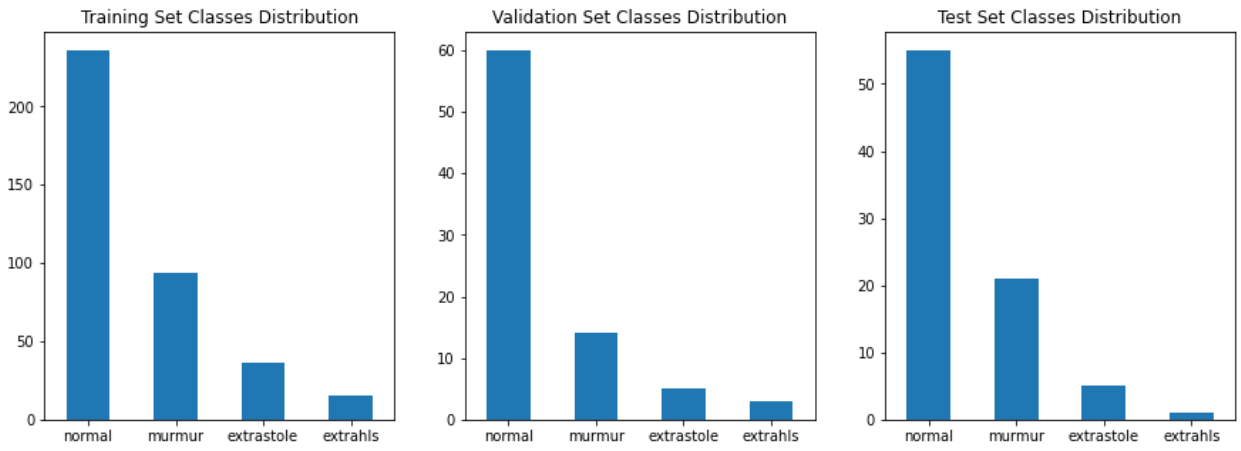

Figure 14: Classification Set Classes Distribution

2.3.2 Dataset Creation

First, we create our Dataset class.

Input should be wav file

Output should Mel_Spectrogram_DB or an MFCC

Steps:

1. We extract audio features as a spectrogram (time as x-axis, frequency as y axis). A spectrogram shows frequencies in linear scale.
2. We convert to Mel Spectrogram which shows frequencies in Mel scale, since humans discriminate lower frequencies better than higher frequencies. (This is done using Mel filters).
3. We transform amplitude into the decibel scale since humans perceive loudness on a logarithmic scale.

During loading of data:

1. We resample wav audio to a fixed sample rate equal to 22.05KHz (like Librosa).
2. We normalize the data so that the sample values are between -1.0 and +1.0.
3. We convert stereo (two channels) to mono (on channel)

The following is an example of a Mel spectrogram in the dB scale (input of our model):

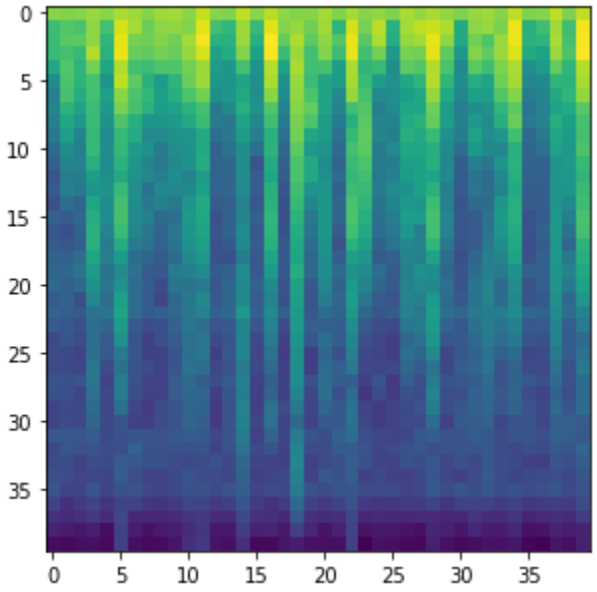

Figure 15: Mel Spectrogram in dB

2.3.3 Custom Model OurCNN

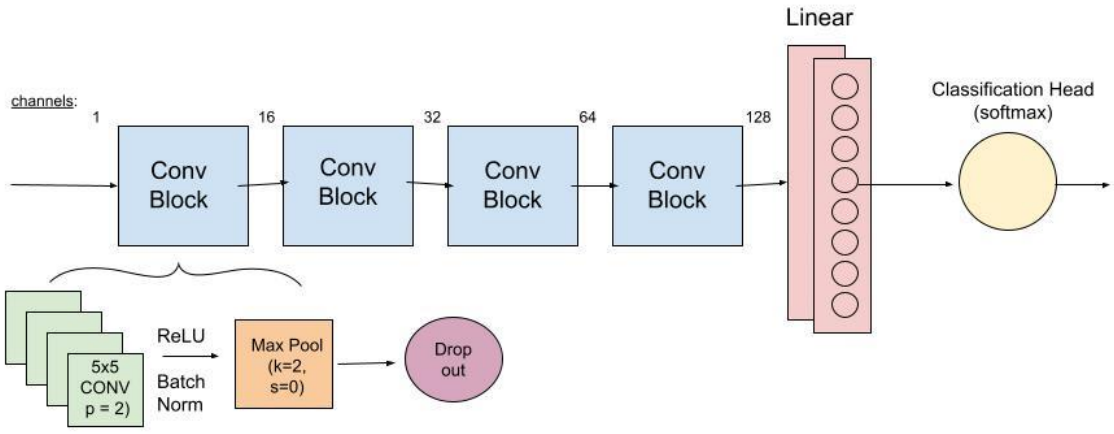

Figure 16: Classification Model OURCNN Diagram

Here, we define a custom model built from 4 convolutional blocks followed by a linear layer (after flattening) then a SoftMax layer to compute predictions.

Each convolutional block consists of a (5x5) CONV layer, a ReLU activation function, and a max pooling layer with a kernel size of 2 (halving).

After each Conv Block there is a dropout layer for regularization to prevent overfitting.

Hidden channels of the model start as:

1x16 -> 16x32 -> 32x64 -> 64x128

Input to this model is either a Mel Spectrogram or an MFCC.

2.3.4 Custom Model: OurCNN_2

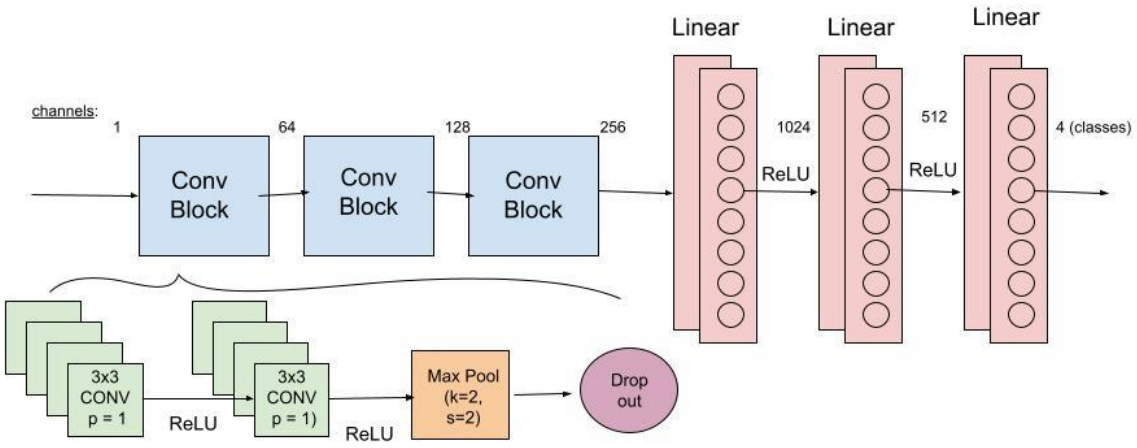

Figure 17: Classification Model OURCNN2 Diagram

Here, we complicate our architecture a bit. Each CONV block is now composed of two convolutional layers instead of one, and we have 3 fully connected layers at the end.

2.3.5 Results and Experiments

We denoised the signals and trained using the best combination of hyperparameters that and the result did not improve a lot.

ID	Model Arch	Denoising	LR	Batch Size	Epochs	Data Type	Loss_fn	Val_F1 (torchmetrics, sklearn) *	Val_ACC (t, sklearn) *	Val_F1	Val_ACC	Test_F1	Test_ACC
1	OurCNN	F	5.00E-04	64	650	Mel Spec	Cross Entropy	0.725 (e) / 0.838 (e) / 0.886 (s)	0.7926 (e) / 0.828 (s)	0.822	0.828	0.7308	0.734375
2	OurCNN	F	1.00E-03	64	450	Mel Spec	Cross Entropy	0.74501 (e) / 0.8386 (e) / 0.886 (s)	0.804 (e) / 0.843 (s)	0.8373	0.84375	0.7431	0.7656
3	OurCNN_2	F	5.00E-04	64	300	Mel Spec	Cross Entropy	0.725 (e) / 0.8608 (e) / 0.886 (s)	0.817 (e) / 0.859 (s)	0.8437	0.85937	0.775674	0.7968
4	OurCNN_2	F	1.00E-03	64	450	Mel Spec	Cross Entropy	0.7412 (e) / 0.843 (e) / 0.886 (s)	0.817 (e) / 0.843 (s)	0.831473	0.828125	0.697837	0.6875
5	ResNet34 (p)	F	5.00E-04	64	450	Mel Spec	Cross Entropy	0.671 (e) / 0.831 (e) / 0.878 (s)	0.78 (e) / 0.8125 (s)	0.8067	0.8125	0.731809	0.75
6	OurCNN_2_Denoise	T	5.00E-04	64	650	Mel Spec	Cross Entropy	0.714 (e) / 0.864 (e) / 0.890 (s)	0.817 (e) / 0.84 (s)	0.714	0.8171	0.703946	0.73475

References

1. Astrophysics data system (no date) NASA/ADS. Available at: https://ui.adsabs.harvard.edu/

2. Audio classification method based on machine learning | IEEE conference ... (no date). Available at: https://ieeexplore.ieee.org/document/8047110

3. Aziz, S., Ahmed, S. and Alouini, M.-S. (2021) ECG-based machine-learning algorithms for Heartbeat classification, Nature News. Nature Publishing Group. Available at: https://www.nature.com/articles/s41598-021-97118-5

4. Cepstrum and MFCC (no date) Aalto University Wiki. Available at: https://wiki.aalto.fi/display/ITSP/Cepstrum%20and%20MFCC

5. Diagnosing a heart attack (2022) www.heart.org. Available at: https://www.heart.org/en/health-topics/heart-attack/diagnosing-a-heart-attack

6. Electrocardiogram signal analysis - an overview - ijcaonline.org (no date). Available at: https://research.ijcaonline.org/volume84/number7/pxc3892826.pdf

7. King, E. (2016) Heartbeat sounds, Kaggle. Available at: https://www.kaggle.com/datasets/kinguistics/heartbeat-sounds

8. Nair, P. (2018) The dummy's guide to MFCC, Medium. prathena. Available at: https://medium.com/prathena/the-dummys-guide-to-mfcc-aceab2450fd

9. Sharma, S. (2021) Audio classification: Introduction to audio classification, Analytics Vidhya. Available at: https://www.analyticsvidhya.com/blog/2021/06/introduction-to-audio-classification/

10. What is a spectrogram? (no date) Pacific Northwest Seismic Network. Available at: https://pnsn.org/spectrograms/what-is-a-spectrogram