2020

Speech Mode Classification using the Fusion of CNNs and LSTM Networks

Pratyusha Chowdary Vakkantula
pv0009@mix.wvu.edu

Speech Mode Classification using the Fusion of CNNs and LSTM Networks

Pratyusha Chowdary Vakkantula

Thesis submitted to the
Benjamin M. Statler College of Engineering and Mineral Resources
at West Virginia University
in partial fulfillment of the requirements for the degree of

Master of Science
in
Electrical Engineering

Thirimachos Bourlai, Ph.D., Chair
Natalia Schmid, Ph.D.
Yuxin Liu, Ph.D.

Lane Department of Computer Science and Electrical Engineering

Morgantown, West Virginia
2020

Keywords: Audio Classification, Spectrograms, Mel Spectrograms, Deep Neural Networks, Machine Learning Classifiers

ABSTRACT

Speech Mode Classification using the Fusion of Machine Learning and Deep Learning based Classifiers

Pratyusha Chowdary Vakkantula

Speech mode classification is an area that has not been as widely explored in the field of sound classification as others such as environmental sounds, music genre, and speaker identification. But what is speech mode? While mode is defined as the way or the manner in which something occurs or is expressed or done, speech mode is defined as the style in which the speech is delivered by a person.

There are some reports on speech mode classification using conventional methods, such as whispering and talking using a normal phonetic sound. However, to the best of our knowledge, deep learning-based methods have not been reported in the open literature for the aforementioned classification scenario. Specifically, in this work we assess the performance of image-based classification algorithms on this challenging speech mode classification problem, including the usage of pre-trained deep neural networks, namely AlexNet, ResNet18 and SqueezeNet. Thus, we compare the classification efficiency of a set of deep learning-based classifiers, while we also assess the impact of different 2D image representations (spectrograms, mel-spectrograms, and their image-based fusion) on classification accuracy. These representations are used as input to the networks after being generated from the original audio signals. Next, we compare the accuracy of the DL-based classifies to a set of machine learning (ML) ones that use as their inputs Mel-Frequency Cepstral Coefficients (MFCCs) features. Then, after determining the most efficient sampling rate for our classification problem (i.e. 32kHz), we study the performance of our proposed method of combining CNN with LSTM (Long Short-Term Memory) networks. For this purpose, we use the features extracted from the deep networks of the previous step. We conclude our study by evaluating the role of sampling rates on classification accuracy by generating two sets of 2D image representations – one with 32kHz and the other with 16kHz sampling. Experimental results show that after cross validation the accuracy of DL-based approaches is 15% higher than ML ones, with SqueezeNet yielding an accuracy of more than 91% at 32kHz, whether we use transfer learning, feature-level fusion or score-level fusion (92.5%). Our proposed method using LSTMs further increased that accuracy by more than 3%, resulting in an average accuracy of 95.7%.

I dedicate my thesis to my family

Acknowledgements

First of all, I would like to express my deepest gratitude to Dr. Thirimachos Bourlai, my advisor and committee chair for always supporting me and guiding me since I began my journey with our lab (Multi-spectral Imagery Lab) at WVU. I thank you for encouraging me each and every time both academically and morally. Your positivity and support have inspired and driven all of us to work towards our goal.

I also would like to sincerely thank the committee members, Dr. Natalia Schmid, and Dr. Yuxin Liu for their valuable suggestions and feedback.

I am always grateful to my family for believing in me and supporting me in every possible way. I would not have been here, both in terms of my career and as a person if it were not for them.

I would like to thank all my labmates for helping me whenever needed and for maintaining such a friendly atmosphere in the lab. I feel extremely lucky to have met my friends in WVU. I cherish all the fun times we have spent together for a lifetime. I thank you all for cheering me up whenever I feel low and I know for a fact that we are going to be this family of friends forever. My time in Morgantown would not have been the same without you.

Finally, I thank WVU and Statler college at WVU for helping us financially through merit waivers, creating a peaceful environment to study and providing us with all the resources. I am glad that I took the decision of joining WVU because it has not only helped me in shaping my career but also given me an opportunity to meet different people. The interactions and friendships with people from different countries made me learn about various cultures of the world. I also thank WVU for keeping us motivated with all the fun activities through Up All Night, ISSS trips, Game days, Student Rec Centre and all the events that I participated in. I take pride in being a member of the mountaineer family.

Table of Contents

List of Figures

List of Tables

Acronyms

AI	Artificial Intelligence
ML	Machine Learning
DL	Deep Learning
ANN	Artificial Neural Networks
DNN	Deep Neural Networks
CNN	Convolutional Neural Networks
RNN	Recurrent Neural Networks
DBN	Deep Belief Networks
LSTM	Long Short-Term Memory
MFCC	Mel frequency Cepstral Coefficients
CRP	Cross Recurrence Plots
ROC	Receiver Operating Characteristic curve
HMM	Hidden Markov Model
GMM	Gaussian Mixture Model
TDSN	Tensor Deep Stacking Network
PCA	Principal Component Analysis
LDA	Linear Discriminant Analysis

1 Introduction

Over the past decade, the fields of Artificial Intelligence [1] and Machine Learning [2] have been growing rapidly. Machine learning is an application of artificial intelligence which uses algorithms to automatically learn tasks by extracting features from raw data so that it can be represented as a model. This trained model can then be used to make predictions on new data. Figure 1 shows the relationship between the fields - Artificial intelligence, machine learning and deep learning.

Machine Learning is widely used for the classification of images [3] and there are plenty of classifiers [4] to do the job. Over the years, machine learning practitioners worked on developing plethora of image classification networks. In image classification, classifiers are given the input in the form of images and trained to extract features in order to classify the images to their corresponding classes.

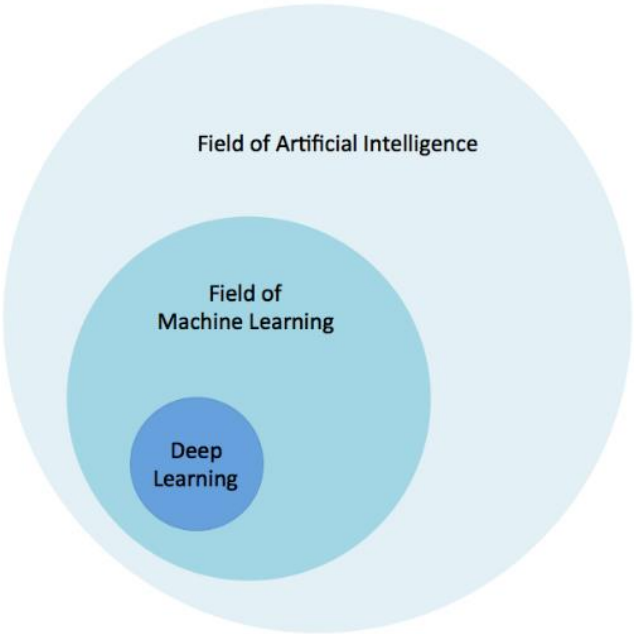


Figure 1. Relationship Between AI, ML and DL

Source: [5]

Recently, sound or audio classification [6] has been of keen interest for data scientists. Most of the work in sound classification area concentrates on environmental sounds [7], urban sounds [8], music genre classification [9], speaker identification [10] etc. There are also a few publicly available audio datasets for environmental sounds, music, urban sounds etc.

Many researchers have developed algorithms and networks that can classify sounds by giving a small audio sample as the input. Features extracted from sound samples like MFCCs [11] can be used as inputs for the classifiers. In addition to these input forms, images are being used to classify sounds, which are proved to be giving promising results in latest times. For this, audio samples are converted to their respective image representations such as – audio waveforms [12], spectrograms [13], mel spectrograms [14], chromagrams [15], cross recurrence plots [16] etc., and these images will be the input for the image classification networks.

While representing audio samples as above-mentioned time or frequency domain images, sampling rate or sampling frequency, window size can play a key role. There is some work done focusing on these aspects. Several pre-trained deep networks [17] that are used for image classification like AlexNet, GoogleNet, ResNet, SqueezeNet etc., are trained for audio classification using images by many researchers.

1.1 Deep Learning

Deep Learning [18], as we all know, is a subfield of machine learning. In the case of machine learning, computers are taught through algorithms to process and learn from the data, and gain experience to be applied on new data, whereas, in deep learning, the computer trains itself by observing and learning the data, thus, getting better by more and more training data.

Artificial Intelligence (AI) is an imitation of human brain. As our brain have neurons to observe, learn and experience things that help us in making decisions, AI or machine learning has Artificial Neural Networks (ANNs). These ANNs learn various features of the data and make decisions just like a human brain.

Artificial neural networks consist of a minimum of three layers – an input layer to receive the data, a hidden layer where the received information is processed and an output layer which decides what to do based on the processed data. A generalized topology of an ANN is shown in the Figure 2 below. Deep learning has the architectures called as deep artificial neural networks or simply, Deep Neural Networks (DNNs). DNNs are basically complex ANNs because DNNs have a greater number of hidden layers, which will help the computer understand more about the input data.

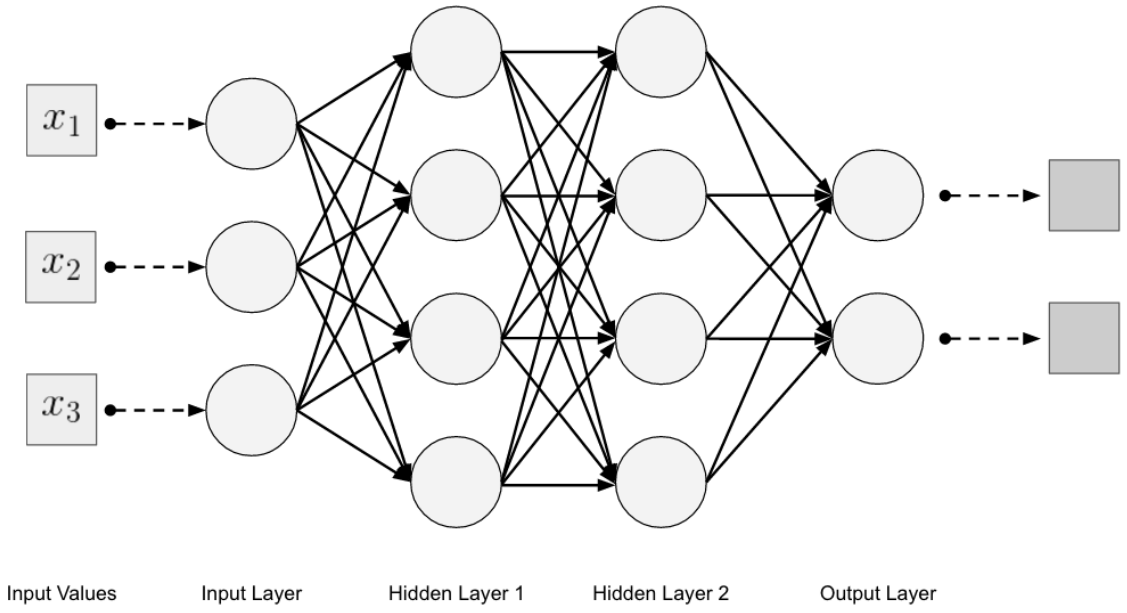


Figure 2. An Artificial Neural Network topology

Source: [5]

Deep neural networks have sub-classes, mainly Convolutional Neural Networks (CNNs) [19], Recurrent Neural Networks (RNNs) [20] and Deep Belief Networks [21] based on architectural differences. CNNs are mostly in visual imagery analysis and the architecture of a CNN is based on shared weights and translation invariance characteristics. These are applied in image analysis like classification, video recognition etc. In RNNs, connections between nodes form a directed graph along a temporal sequence, which exhibits temporal dynamic sequence. Their internal state memory is used to process variable length sequence of inputs. Unsegmented tasks such as

handwriting recognition, or speech recognition uses RNNs. DBNs have multiple layers of hidden units (latent variables). It has connections between the layers but, none between units within each layer. These are used in classification tasks. These deep learning techniques [22] [23] [24] [25] [26] [27] are used in various ways by researchers for image and sound classification. They are being applied in vast number of fields like computer vision, speech recognition, image analysis, audio classification etc.

1.2 Motivation

Classification is one of the most important and common techniques applied in artificial intelligence. There is a plethora of classifiers in this field, that can be used to perform the task. Of everything, classification using images has been a dominating approach to classify various objects since the beginning of the development of artificial neural networks. Images are read in terms of pixels by a computer and thus learn different features of the images like edges, blobs, corners, SIFT, SURF etc., and use them to classify the given image into its corresponding class.

After deep learning made its way to evolve, deep neural networks are being used to classify images. Deep networks are known for their accuracy and great performance for large amounts of input data. So, image classification has been taken to next level with the advancement of deep learning. Many pre-trained networks like AlexNet, GoogleNet, ResNet, VGG, Inception, SqueezeNet etc., are developed and are available online. These pre-trained architectures have given 100% accuracy in many studies in the past. All these works prove that there are sufficient number of image classification techniques that gives reliable results to be applied in user identification, crime solving etc.

The problem is it is not visual data all the time. In other words, we might be having different forms of input data other than images or videos. For example, audio. Recently, audio classification is being talked about and worked on. Audio data can be very crucial in identifying a person or catching a criminal as well. This gained the interest of many researchers to focus on audio and develop audio classification techniques.

Like images, features can be extracted from audio files and these extracted files are given as the input to train a classifier. Later, when given the new data, the classifier will be able to classify it. In order explore audio classification, a few researchers thought of converting audio to images.

Because these images can then be used on image classification algorithms. For this, audio files are represented in the form of images, like spectrograms, and then the same procedure for regular images will be followed. Classifying sounds using spectrograms has gained popularity so, pre-trained deep networks are also being used to classify audio using their image representations. Satisfactory results are seen using pre-trained networks in previous works.

Most of the work related to sound or audio classification so far has involved classification of environmental sounds, urban sounds, music. Some work was done on speech sounds for speech recognition and emotion classification. Not much work was done using humans talking style categories as per our knowledge. This gap in sound classification has motivated this thesis to fill in.

1.3 Research questions

Question 1: Is the new data used in this work going to affect the performance of the deep networks in a negative way?

Motivation: Dataset in this work is not like those used in most of the other works of sound classification like ESC-50, ESC-10, UrbanSound8K, or other music and speech datasets. Our audio files are collected from YouTube and some of the files have no noise where on the other hand, a few others have a little, which makes it a semi-controlled dataset, so, the types of classes are different from other common audio datasets that are available online. So, how this new data is going to work with the pre-trained deep networks is going to be a key question to answer.

Question 2: Is it better to use spectrograms and mel spectrograms separately or fuse both of them to be given as inputs?

Motivation: Spectrograms and mel spectrograms frequency representations in log scale and mel scale respectively and are proven to be the most reliable representations of audio data [28] when it comes to audio classification using deep learning but how much of a good idea is it to fuse spectrogram with its corresponding mel spectrogram in terms of accuracy when compared to using each of them separately is an interesting question to answer through this thesis. By the end of this thesis we should be able to know whether fusing those two is going to improve the results or worsen them.

Question 3: How sampling rates in image representations are going to influence the accuracies?

Motivation: Regarding the role of sampling rates in audio classification, there has already been some work done by other researchers. But, because of the reason that our audio dataset is different, and we are also using fused images, the impact of sampling rate can be a good point to observe as well.

Question 4: How the accuracies will be affected when deep networks and regular classifiers are combined?

Motivation: There was so much work done in sound classification using deep learning techniques and machine learning classifiers separately and compared them. A few researchers tried to combine both methods in different ways like combining at score level, so that the overall accuracy scores will be increased and so on. In this work, we are going to use both by giving the features extracted from deep networks as the inputs to the ML classifiers and compare them with deep networks alone.

1.4 Problem Statement

Unlike environmental, urban sounds, and music data, not much work was done with people's talking styles. In this work, talking styles are categorized into monologue, whispering, chanting. Monologue is like normal talking, whispering is talking with lower pitch than that of monologue (like whispering in one's ear), and chanting is a group or an individual saying out temple chants/mantras. These three classes are selected for this classification problem because they are distinguishable for a human ear in terms of their pitch. More about these classes is discussed in 4.1. Using image classification techniques to classify audio is not completely new and there have been several effective results in case of other popular datasets like environmental sounds, but what happens if it is with this new data is the question that needs to be answered yet with this work. In the past, similar problem has been dealt with in other works by training pre-trained deep networks or regular classifiers. This thesis mainly focuses on the effect of combining deep networks with regular classifiers like SVMs on the test accuracies.

1.5 Aim

To put it simple, we aim to answer all the research questions that were developed, through this thesis. By following a step-by-step process, the agenda of this work is to be able to get desired accuracies and make sure that the combined system of deep networks and classifiers will be reliable to classify speech audio using spectrograms or mel spectrograms. During this process of reaching our goal, we intend to make other observations and conclusions about the best form of input for audio classification, best algorithms to classify etc.

1.6 Research contribution

The work in this thesis contributes to growing research in speech sound classification using deep learning. Several contributions are made to the area by this thesis, that have not been made before.

To begin, data used in this research is unique and has not been used in any other work. All the data (audio clips) was collected from YouTube. Each audio clip of a subject is of 118-125 seconds long and audio of such long duration was never used in any other work in the past to the best of my knowledge.

Secondly, using image representations of the audio clips as inputs is not completely common in sound classification. We chose spectrograms and mel spectrograms in our case and of course, there were papers where such image representations were used previously. But we also fuse spectrograms and mel spectrograms to form a single image and use this fused image as one more form of input to the algorithms. Fusing those two image forms to study the effect is something new in this research.

Thirdly, although there have been several works that were done in the past about the effect of sampling rate on audio classification, this is the first time it is being done in the case of speech modes classification, to the best of our knowledge. Like in the open literature, we used two different sampling rates – 32kHz and 16kHz. According to Nyquist Sampling theorem the sampling frequency to produce the exact original waveform should be double the original frequency of the signal. Since the human hearing bandwidth is 20Hz-20kHz, the audio sampled can be at the rate of around 40kHz.(Usually 44.1KHz is preferred). To study the effect of sampling rates, the rates that fall within this Nyquist rate – 32kHz and 16kHz are chosen. The usage of sampling rates in this work is discussed in 3.6.

Finally, the most affective image classification architectures, deep neural networks are used. Results from the work of previous researchers shows that deep networks are quite reliable for audio classification as well. To contribute to the field, we use pre-rained deep networks and LSTM network to conduct two set of experiments. For a third and most important experiment, we combine these deep networks and classifiers.

Thus, we will conclude what image representation (spectrograms or mel spectrograms or fused) at which sampling rate (16KHz, 32KHz) gives the best accuracies using what algorithm (pre-trained deep network like AlexNet; or LSTM network; or DeepNet-Classifier combination).

1.7 Organization of this thesis

The remainder of this thesis is organized into 5 chapters as follows

Chapter 2 will go through the previous work done in this area by further dividing it into two categories – Non-speech related and Speech related literature review.

Chapter 3 is about data preprocessing and briefly discussing the methods used to perform experiments in this thesis.

Chapter 4 explains the experiments done to study and complete the research. This chapter talks about all the experiments done using different classification methods like deep networks or classifiers including the results obtained.

Chapter 5 concludes the thesis by showing what is implied from the results. How this work can be extended in the future will also be discussed in this chapter.

2 Literature Review

In this chapter, we will discuss about the previous works that were done in the field of Audio/Sound Classification by several researchers and data scientists. We can understand how the work has been progressing in this area since the beginning and the research gaps that needs to be filled.

The chapter is divided into three sections. First section will talk about the literature of work related to sound classification using machine learning algorithms or methods like classifiers - Support Vector Machines (SVMs), K-Nearest Neighbors (KNNs) etc. The second section is about the previous work on classifying sounds using deep learning techniques. The last section specifically discusses the literature of speech mode classification.

Initially, recognition of sounds has a much limiting domain just and it was a difficult task to be able to classify sounds. Because a huge percentage of artificial intelligence field deals with images and the methods used are image classification techniques. Somehow researchers had developed methods and algorithms that can classify and recognize audio files. All these techniques were under the umbrella of Machine Learning field. Yet because of the plethora of image classification or image recognition techniques, which have started to become much more advanced and affective over time due to the development of Deep learning, researchers who are working on audio classification have started to adapt these deep learning techniques to audio by converting original audio samples to be represented in a graphical image like spectrogram, waveform etc. Eventually, classification of audio using deep networks have been proving to outperform machine learning methods alone through profound classification accuracies. In this field of classification, the sounds include environmental, urban sounds like birds chirping, dog barking, honks from vehicles, machine and engine sounds, music like various instruments, music genre classification, and speech sounds like speech recognition, emotion recognition etc. Hence, the literature in this chapter includes the work related to any of the sounds, be it speech or non-speech sounds, and their classification and the methods used to do so.

2.1 Sound classification using Machine Learning techniques

This section will cover the literature related to classification of audio/sounds using machine learning and/or neural network techniques.

Initial research had led to build a program that performs non-speech sound recognition like environmental sounds. The program was developed by the Cornell Lab of Ornithology called, The Canary program, designed to recognize bird song [29]. A given signal is analyzed by the program first and then a spectrum is plotted by the user. Unfortunately, the program failed in identifying a signal as the song of a specific type of bird and required manual work in order to match the sounds.

For feature extraction, Goldhor uses Mel frequency cepstral coefficient (MFCC) technique and a modified vector classification technique, to perform supervised clustering into classes [30]. In this work, mean and variance are also calculated for each sound class, and to make all the samples to be of a constant length, a time warping technique is also used. Goldhor notes that the difficulties with sound identification research may occur in sound separation and different environment issues.

Single impulsive sounds are those that are created by the impact between objects. To classify five of such single impulsive sounds, Hiyane [31] presents a signal processing-based system. Unfortunately, Hiyane's work did not mention the specific technique used to classify the sound features, that are distinguished based on peak and reverberation times. He also observes problems like Goldhor regarding multiple sound segregation and that different sounds produce distinctly different waveforms.

Dorken et al., [32] presents an interestingly unique approach for recognizing environmental sounds. In order to both recognize the sounds and separate them, knowledge-based signal processing methods are used. In this method, comparison against a contour developed from a waveform using short-time Fourier transform (STFT) is done. This is a novel approach that groups both advanced signal processing and sound understanding approaches together using knowledge-based techniques. The main drawback of this technique is that it requires substantial effort in building up, therefore, slow in computing and not so suitable in applications that require fast response times.

Reyes-Gomez and Ellis [33] developed a method that uses cepstral coefficients for feature extraction combined with a clustering technique and Hidden Markov Model (HMM's). The clustering technique or a Gaussian Mixture Model (GMM) is used to combat the lack of natural basic units in HMM's. This technique achieved a classification accuracy of 85% - 90% on their arbitrarily selected classes of sound depending on the clustering technique used. However, the usage of HMM is not fully explored. Also, they have no defined way for their system to make more refined classification, other than using traditional pattern recognition techniques.

Liu presents an LVQ (Learning Vector Quantization) based technique for the recognition of ground vehicles like tanks [34]. Liu uses the standard LVQ algorithm as in the work of Kohonen [35] as well as two modified LVQ techniques - Tree Structure Vector Quantization (TSVQ) and Parallel TSVQ (PTSVQ). The PTSVQ technique gives 90% classification accuracy with sounds that are already trained into LVQ network whereas, with "unknown" test sounds, a recognition rate of 68% is achieved. In addition, Liu did not mention about the number of vehicle classes used and how or why the LVQ technique was selected.

Sampan presents a ground vehicle recognition system [36] where he tests several variations of multi-layer perceptron (MLP) neural networks and fuzzy algorithms. He uses "ideal" dataset, and the performance of all algorithms is close to 100%. In the case of real data, five classes of ground vehicle are taken, and test data is classified into one of those five classes. This test gave around 75% classification accuracy. However, the effect of increment of sounds on the performance is unclear in the work.

Wang et al., [37] presents a Gabor- based non-uniform scale frequency map to classify environmental sounds. Matching pursuit algorithm is used to select the important atoms from Gabor dictionary for each audio frame. The scale and frequency are extracted from the atoms and a scale-frequency map is constructed. At this point, Principal Component Analysis and Linear Discriminate Analysis are applied to the scale-frequency maps to extract proposed feature. Using these features classification of sound samples is performed using Support Vector Machines (SVMs). A high accuracy rate is reported.

Zhang et al., [38] worked on audio segmentation and classification. In this work, five audio classes were used – silence, music, background sound, pure speech and non-pure speech, which further has speech over music and speech over noise. A sound stream is segmented by classifying

each sub-segment into one of the five classes and then evaluated the performance of different classifiers and by comparison concluded that SVMs were the most accurate implementations.

Silva [39] employs Support Vector Machines with Sequential Minimal Optimization (SMO) in his work to classify, segment and chronologically predict cinematic sound. He used probabilistic output from logistic regression to segment fixed length parts into auditory scenes. His proposed method, SMO classifier had shown better results compared to K-Nearest Neighbor, Naïve Bayes and standard SVM classifiers.

Mostafa et al., [40] compared machine learning techniques to standard statistical methods to classify western musical genres in this work. The three main artificial neural networks used by them are multilayer perceptron (MLP), probabilistic neural network (PNN) and self-organizing maps neural networks (SOM) whereas the statistical methods are Linear discriminant analysis (LDA) and Cluster analysis (CA). They used five features – average frequency, variance frequency, maximum frequency, amplitude and median to perform classification. Results proved that ML methods outperformed the statistical methods.

2.2 Sound classification using Deep Learning

In this section, literature review on classification of sound/audio using deep neural networks will be seen. Recently, classifying sounds using deep learning techniques has been of immense focus. To classify audio using deep networks like CNNs, the input data must be in the form of images. So, in all the works that used deep neural networks in order to classify sounds, the original audio clips are converted to image forms like spectrograms, or mel spectrograms, or waveforms that represents the given sound signal as voltage or amplitude over time. Several waveforms are used in various works as an input to different kinds of deep networks and evaluated for best classification accuracy. Deep Learning is a wide area with plenty of techniques, methods, and applications [41].

McLoughlin et al., in their two papers [42] [43] states that the robust sound event classification, where the recognition of various sounds in a real-world noisy condition is a challenging task and proposes that a deep neural network is a reliable solution to the problem. In their work, they compare auditory image front end features and spectrogram image-based features. They used Support Vector Machines, and other state-of-the-art deep network

classification techniques and stated that best accuracies occurred with spectrogram image features.

Karol Piczak [44] concentrated on seeing if CNNs can be used to classify environmental sounds. He developed a basic deep model with two each of convolutional layers and fully connected layers. He used segmented spectrograms as the input and studied that his model outperformed other baseline and state-of-the-art techniques. Along with spectrograms, the deltas were computed and fed into the network in two channels. Since there is limited data available of environmental sounds, augmented the training sound samples by adding random delays and class dependent time stretching to the original recordings of ESC-50 & ESC-10 [45] and UrbanSound8K [46] datasets.

Boddapati et al., [28] applied image classification techniques like deep networks to classify environmental sounds. In this work, the three popular environmental data sets ESC-10, ESC-50 and UrbanSound8K were trained on AlexNet [47] and GoogleNet [48]. They converted the sounds to spectrograms, MFCC and Cross-recurrence plots (CRP) images with varying sampling rates. A set of experiments were also performed by fusing the three image formats (spectrograms, MFCC, CRP) into a single image and giving it as the input. Results have shown that, the sampling rates of the best classification accuracies were different for each data set Convolutional Recurrent Neural Networks were also trained but the results were not that satisfactory.

Salamon et al., [49] focused mainly on data augmentation in their work. They proposed a Deep neural network with 3 convolutional layers. They applied four data augmentations – time stretching, pitch shifting, dynamic range compression, background noise on Urbansound8k dataset and generated five sets. It was evaluated that the combination of augmented data and the proposed CNN gave comparable state-of-the-art results. Also, the effect of each augmentation on the accuracies were studied and it was suggested that class-conditional data augmentation can be applied to improve the performance further.

Shawn Hershey et al., [50] have compared a few state-of-the-art DNNs like AlexNet, VGG, Inception V3, ResNet-50 with their baseline fully connected DNN. They used YouTube-100M data set that consists of 100 million YouTube videos and also AudioSet [51]. The main motto of their work is to study the impact of the size of training data on the classification rate. Results suggest that training on larger label set vocabularies can improve performance.

Kons et al., [52] presented their work by classifying outdoor audio events. As a main classification method, they used a Deep Neural Network and compared the results with SVM and Gaussian Mixture Model (GMM) algorithms. GMMs are a probabilistic model for representing normally distributed subpopulations within an overall population. In this work, they used audio from freesound.org, which is an open source repository for users to upload or download wide range of audio events. They proposed a new method to improve the process of pre-training by introducing scaling factors. The results have shown that the performance of a DNN is better and comparable to SVM, but the accuracy of GMM is very poor. To improve the results, they fused both DNN and SVM at score level and a new score is generated, which increased the score by 6.7% when compared to DNN alone.

Aditya Khamparia et al., [53] proposed a Convolutional Neural Network (CNN) and a Tensor Deep Stacking Network (TDSN). The TDSN consists of multiple, stacked blocks, where each block contains a bilinear mapping from two hidden layers to the output layer, using a weight tensor to incorporate higher order statistics of the hidden binary (0, 1) features. They have used the data sets ESC-50 and ESC-10 and generated spectrograms of those original recordings. These spectrograms were given as inputs to CNN, TDSN and the results were compared to those of Machine Learning classifiers like SVMs, Decision trees, KNN, Random forest, Multilayer perceptron. They observed that their proposed architectures gave better results and also reduced the number of trainable parameters using spectrograms as compared to direct sound classification.

2.3 Speech mode classification

Unlike the cases of environmental sounds, music genres, and speaker identification, there has not been much work done on speech mode classification. In their work, J. B. Wilson et al., [54]studied the processing of adult male and female whispering and normal speech using the Locally Predicted Coder, known as the LPC-10 vocoder. This work also talks about the substantial increase in the formant frequencies when shifting from phonated to whispered speech. Siobodan et al., [55] discussed in detail about formant features in their work. They used five Serbian vowels of both male and females to investigate the formants. Spectrograms were used in Stanley's work [56]. They developed a unique process of calculating magnitudes of the energies of normal and whispering sounds in high and low frequency bands and thus the results are based on magnitude ratio. Zhang et al., [57] performed automatic speech classification using GMM and developed a speech mode classification technique by classifying five classes of speech modes. They also studied about frame

energy distribution and speech intensity. Zeynab et al., [58]used an LSTM (Long Short-term Memory) network trained on log-filterbank energy (LFBE) acoustic features to develop a whisper speech detector system. These authors compared LSTMs with MLPs (Multi-Layer Perceptron) and made additional comparisons by adding six more signal features.

Summary

In this chapter, we discussed the literature of sound classification using machine learning techniques, deep learning techniques and lastly on speech mode classification. From the discussion we infer, a lot of work has been done in the field of sound classification regarding environmental sounds, urban sounds, music, speaker identification using speaking sounds etc. But comparatively, very less research could be found in speech mode classification. Moreover, the application of deep learning techniques/algorithms to speech mode classification is found to be very limited. We obtained our motivation from this gap in the literature, which we intend to fill through our work.

3 Methodology

In this chapter, we go through the methods or techniques used to set up and implement the experiments. A brief introduction or definition of each topic that we used in this work is given and explained, if necessary.

3.1 Introduction to Data Preprocessing techniques

Transforming raw/unstructured data into understandable format so that, it is ready to be fed as input to the algorithm is called preprocessing the data. Data preprocessing is a data mining technique to convert the unstructured data into clean or structured data. Today, because of the huge availability of internet, data can be collected from heterogeneous sources. Unstructured data is nothing, but the data collected from different platforms and is of various formats and will not be ready to train an algorithm unless a few changes are made.

Most of the times, data in real world is incomplete, inconsistent, and/or lacking in certain attributes, and noise. If there is so much unreliable or redundant information present in the raw data, then it will be difficult for an algorithm to learn and extract features from it. This will result in bad accuracies. So, it is important that the data we use is well structured because it plays a vital role in the outcomes of the experiments. Each algorithm, if not unique, needs data to be in a certain format. It is always beneficial to have the data in a format such that, it can be used for more than one algorithm like both machine learning and deep learning, which allows us to choose the best one among them.

The quality of a dataset can be assessed in three main factors:

- Accuracy: Humans tend to make errors often. These errors might result in the inaccuracy of the data. Some examples of such errors are like putting the data that has to be in one class into another, duplication of data, entering incorrect values or numbering etc.

- Consistency: Aggregation of data is always inconsistent. Collecting or gathering data from different sources might result in the data not being uniform or of not same format. Files of different formats cannot be given to any algorithm for training.

- Completeness: Sometimes dataset will be lacking attributes of interest, or features. Also, the data we look for will not always be available, at least in the quantity. These factors result in incomplete data.

To ensure the above three factors are present in our dataset and wish to get acceptable results from our algorithms, it is highly recommended and required to preprocess the data. So, data preprocessing is divided into four parts – Data cleansing, Data editing, Data reduction, and Data transformation

- Data Cleansing: This process is done by filling in missing values, smoothing the noisy data etc. We first try to identify the incorrect or corrupt parts of the collected data and then replace them with correct values, and/or deleting or modifying them if needed. Each task is preformed using different techniques. For instance, sometimes, labels will be missing for a few training examples. In that case of missing data, we can simply ignore the training example by deleting it, or if there are many such missing labels then we can replace them with 'unknown' or something like that, or use mathematical measures of central tendency (mean, mode, median) to replace them, or manually enter the value, if possible. To remove noisy data, methods like linear regression can be used to smooth out the noise, or to detect outliers, approaches like clustering are used.

- Data Editing: In this step, problems with having different formats or representations will be resolved. When data is collected from different sources, there is a high or maximum possibility that the data is not uniform, whether it is in terms of file type, or size etc. So, dealing with these types of issues is done either manually or with the assistance of a computers or sometimes a combination of both. Techniques in this step include file resizing, converting files from one type to another so that, all the data is uniform.

- Data Reduction: This step is to achieve a condensed representation of a dataset, which will be a smaller version of the original in terms of volume, while also maintaining the integrity. This step helps in having a smaller dataset yet yields similarly efficient results. Data reduction can be done by deleting those training samples with missing labels, that cannot be corrected with data cleansing techniques, or with noisy information whose noise cannot be reduced. Also, low pass and high pass filters can be used to remove those normalized attributes that have distribution less than or more than a threshold. PCA (Principal Component Analysis) is a popular statistical method used to reduce the number of attributes using correlation since correlated attributes follow similar trends.

- Data Transformation: This is the step where the corrected, reduced raw data will be transformed to another format to feed to the algorithm. The strategies used at this stage are smoothing, adding, or constructing new attributes (features), if needed, applying normalization to the data, augmentation methods etc. Transformations of data are applied to particular entities like rows, columns, data values, fields etc., includes actions like parsing, standardizing, joining and so on. These data transformation actions are mainly done using spreadsheets such as Excel.

Every dataset is different and unique. So, it is not always mandatory to follow every step of data preprocessing mentioned earlier. In some cases, we might not need data reduction whereas, in others, we will not be needing data editing etc. Thus, data preprocessing has several approaches to deal with and a researcher should be aware of all the preprocessing techniques that are available and be able to find out what is required for his data in order to obtain better outcome since data plays a key role in results.

Dataset used in this thesis was collected from YouTube and is not used in any other works. It is also first of its kind, which makes our data, the pillar of our work. So, in the following sections, classes and the various input formats used will be explained in detail.

3.2 Audio waveforms

A waveform is said to be a one-dimensional graphical representation of an audio signal that is displayed as a function of time. It is not an image in its original form but represents changes in amplitude over a period of time. It is a pressure wave, which is converted to an electrical signal (such as voltage) and displayed as a function of time. In general, X-axis (horizontal axis) is used for time whereas Y-axis (vertical axis) measures the amplitude. The idea behind an audio waveform is to give us a visual clarity of how the audio file is and what has been recorded. If the vertical lines (amplitude) look smaller, it means the audio is quieter, on the other hand, if the amplitude spikes look bigger or longer, then the audio file at that particular time is louder. Typically, a waveform contains of several number (thousands) of discrete changes for a short period of time, but when you zoom in, you would be able to see its contour in more detail, which is why the concept of waveforms is considered to be abstract.

In general, a waveform is a digitized recreation of dynamic changes in voltage with respect to time. There are four types of waveforms that are known to be the basic ingredients to construct any audio waveform. These are also called synthetic waveforms as they can be synthesized without an audio clip. They are sine wave, square wave, triangle wave, and sawtooth wave.

- Sine wave: It is the simplest of all waveforms that contains only a single fundamental frequency and does not have any harmonics. Fundamental frequency is used to determine the pitch of the sound. Adding harmonics and overtones makes it distinguishable for each sound.

- Square wave: A square wave is a little more complex when compared to a sine wave because of the odd harmonic content in it. Unlike sine wave, its envelope is in square shape and it seems like there is no smooth transition in terms of amplitude, and changes instantly from minimum to maximum amplitude or vice versa, when we observe visually.

- Triangle wave: A triangle wave contains fundamental sound and odd harmonics and can be compared with the square wave. It will be evident from the graphical representation of a triangle wave that the power of each harmonic in it is twice as low as those in a square

wave, which is why the power in a triangle wave is reduced twice as fast as that in a square wave.

- Sawtooth wave: This waveform also contains both even and odd harmonics and is known to be the richest in terms of timbre. Timbre is the perceived sound quality of a particular musical note or tone. It is the sound quality that helps our ears to distinguish sounds with same pitch and loudness.

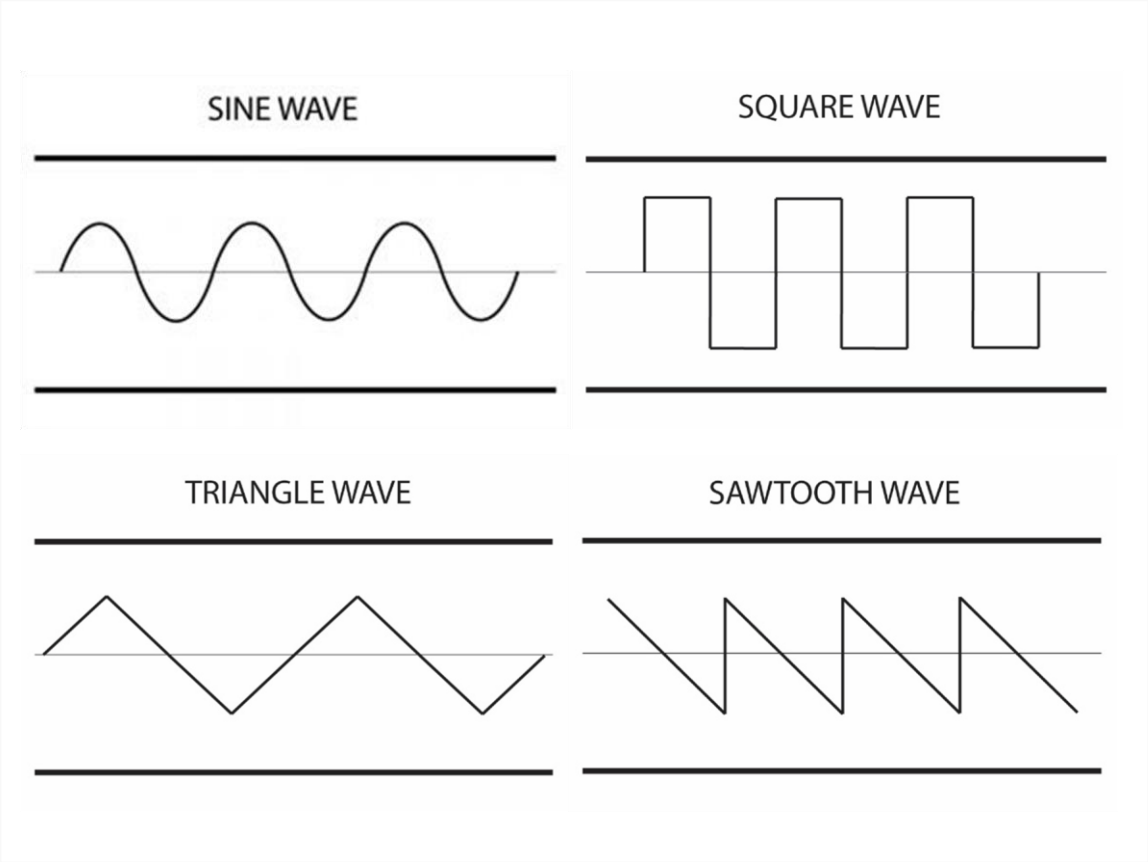


Figure 3. Types of basic waveforms

3.2.1 Generation of audio waveforms

With time on X-axis and amplitude on the Y-axis, audio waveforms are generated. In this work, audio waveforms are generated for all three classes – chanting, monologue and whispering. For each class, two sampling rates are used, which are 16KHz and 32KHz. Hence, for each class, two

sub-groups are created by naming in terms of sampling rates for audio waveforms. An example of how a waveform looks like is shown in the figure below.

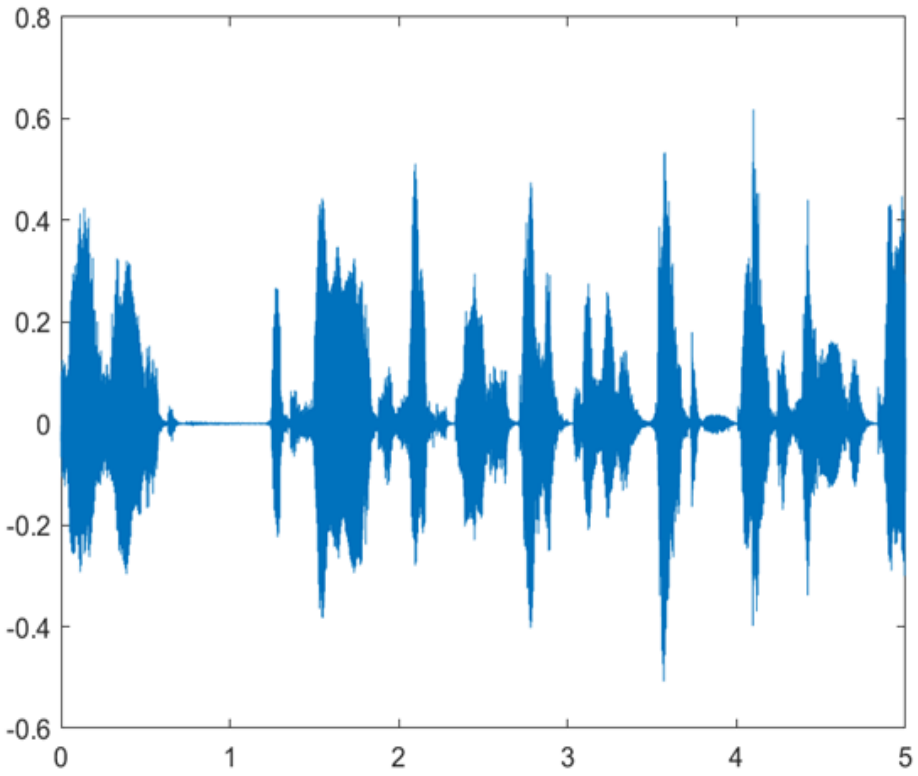


Figure 4. Example of an audio waveform

3.3 Spectrograms

A spectrogram is a visual representation of a spectrum of frequencies. They represent the loudness or strength of the signal over time at various frequencies. Spectrograms are widely used to display sound wave frequencies produced by either humans or any other machines, environmental sounds etc. The frequency content of the spectrograms allows us not only to see if there is more or less energy at a given time but also how these energy levels vary with respect to time.

Spectrograms are two dimensional, but there is a third dimension which is represented by colors. When the spectrograms are shown as 3D plots, they are also called as waterfalls. On

horizontal axis, time runs from left to right, and frequency is measured on vertical axis. This frequency measurement is interpreted as tone or pitch of the sound. Third dimension is used to measure the amplitude of the signal using colors, which represents the loudness. There are several colormaps available with different color combinations. By default, they are represented in Parula colormap, in which, dark blue color is for lower amplitudes and, brighter colors are for louder regions.

3.3.1 Generation of Spectrograms

Spectrograms can be generated in two ways – using a filterbank with a series of bandpass filters or calculate using Fourier transform from the time signal. Bandpass filters method is analog processing method in which, the input signal is divided into frequency bands and a transducer will be controlled by each filter's output magnitude. This transducer records the spectrogram as a graphical image on the paper. Using Fourier transform or Fast Fourier Transform (FFT) is a digital method where, in the time domain, the digitally sampled data is divided into chunks (these chunks can overlap), then the Fourier transform is applied for each chunk to calculate the magnitude of the frequency spectrum. Each of these chunks are then corresponded to a line in an image and all

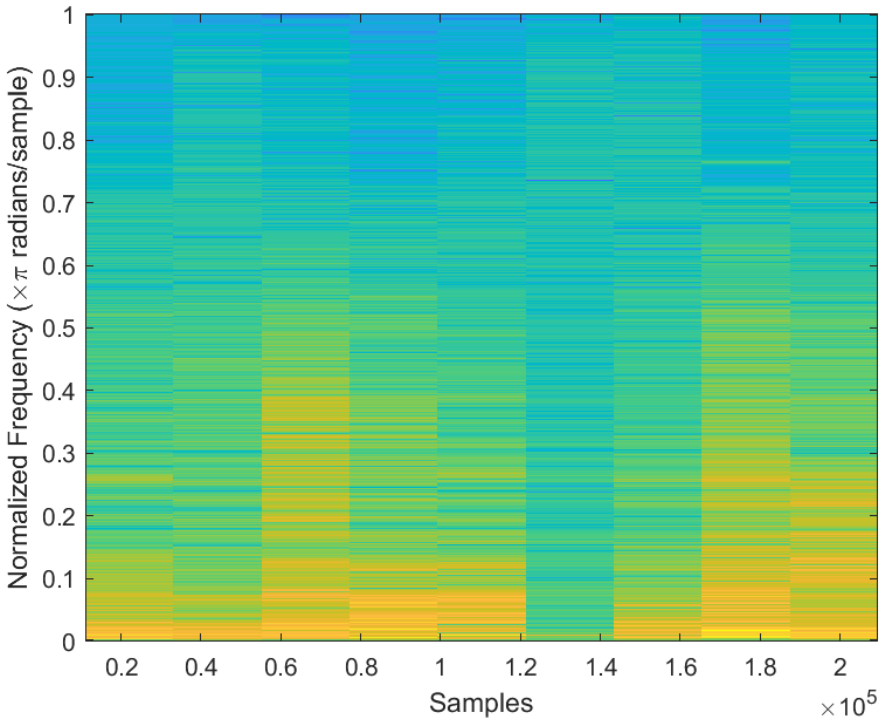


Figure 5. Example of a spectrogram

the lines or time plots are placed side by side to form an image, which is nothing but the spectrogram. To be specific, Short-Time Fourier Transform (STFT) is used to generate spectrograms. STFT is simply a sequence of FFTs of windowed data segments, where the windows are allowed to overlap in time, usually by 25-50%.

Since a spectrogram is a visual representation of how the frequency content of a signal changes over time, time is represented along the X-axis, frequency along the Y-axis, and amplitude or energy level of the signal at a particular time and frequency is represented along the third dimension, which is the Z-axis.

In our work, spectrograms are generated for all the three classes at 8KHz, 16KHz and 32KHz sampling frequencies. Example spectrograms are shown in the figure below.

3.4 Mel-spectrograms

To put it in simpler words, a mel spectrogram is nothing but a spectrogram in which the frequencies are converted to the mel scale. In order to fully understand the definition of mel spectrogram, we need to be aware of what mel scale is.

- Mel scale: Mathematically, a non-linear transformation of the frequency scale results in the mel scale. The purpose of this mel scale is to be able to interpret the differences in the sound signals even at higher frequencies. This is because humans are generally capable of perceiving or telling the differences between two frequencies if they are at a lower level. But it is extremely difficult to catch the differences between two signals if they are at higher level of frequencies. For example, it is easy to tell the difference between 400Hz and 1000Hz, but we can barely notice the difference if 12000Hz and 12600Hz is the case. To solve this issue, a unit of pitch was proposed so that the equal distances in pitch sounds equally distant to the listener irrespective of the frequencies. This unit of pitch is called as mel scale.

- Mel spectrogram: As mentioned above, a mel spectrogram is a spectrogram that has its frequencies in mel scale. It is also defined as a representation of the short-term power spectrum of a sound signal just like a spectrogram. The only difference is that the mel spectrogram has mel scale as its y-axis unlike a spectrogram that has the log scale of frequencies. An example of a mel spectrogram is shown in Fig 6.

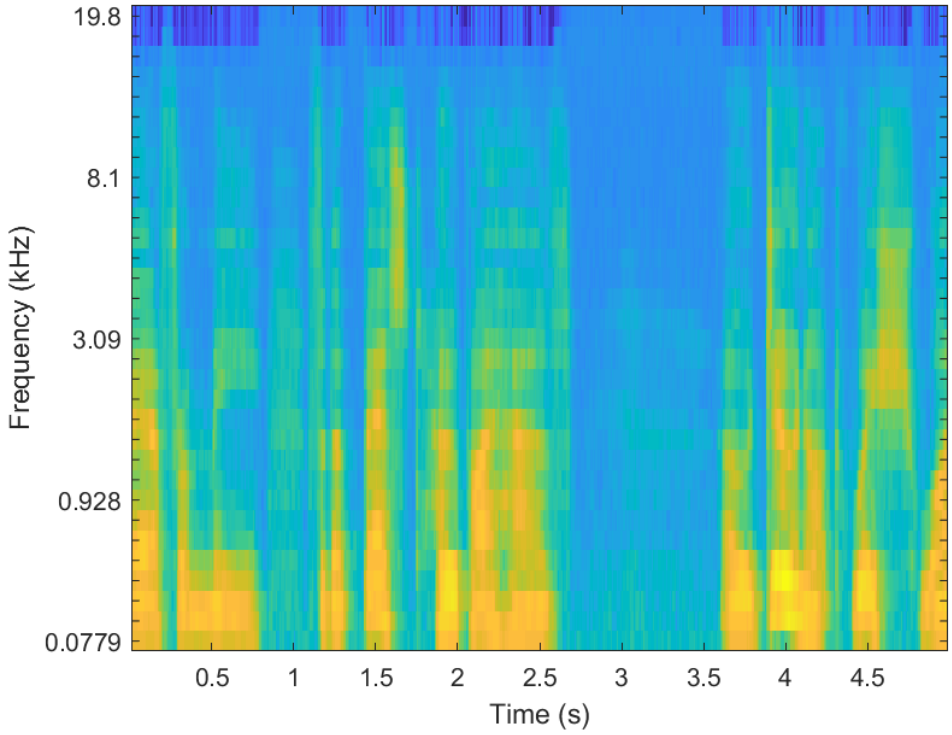


Figure 6. Example of a mel spectrogram

3.5 Fusing spectrograms and Mel-spectrograms

Fusing of images generally is the process where two images are superimposed on one another to generate a third image. Thus, a fused image is the resulting or the output image of performing some kind of operation on two input images in order to combine them to form a single image. These operations can be of several types due to which there are many methods in which the process of fusing/superimposing of two images can be done. Of all those, we briefly discuss here the five

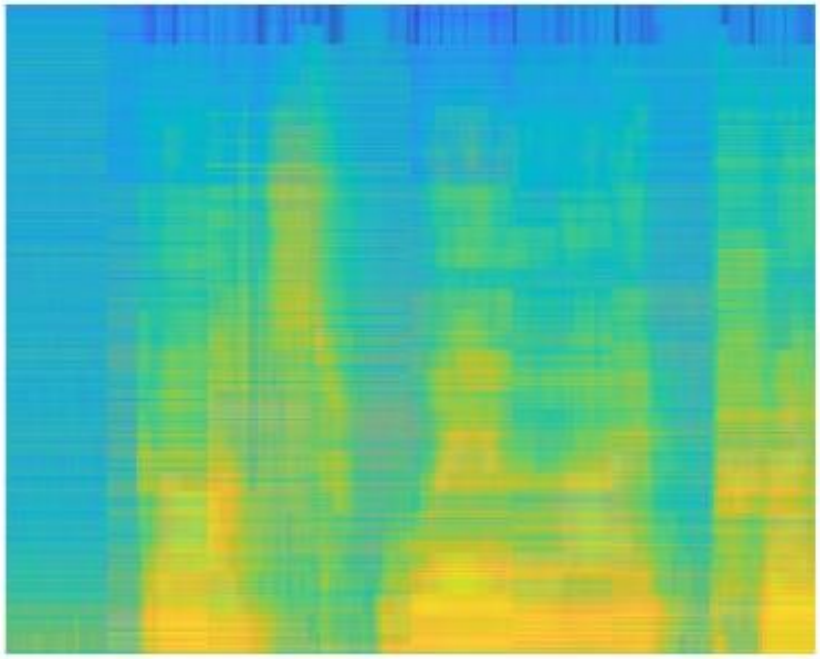


Figure 7. Example of a fused image (fusing spectrogram and mel spectrogram)

methods that are used in MATLAB since it is the software that we used in this thesis. The names used below to define the method are exactly how they are used in the MATLAB function.

- 'falsecolor': In this method, the composite of two input images is performed and the two images are superimposed in different color bands. The regions where the two input images have same intensities, then the composite image shows gray color whereas in the regions where the intensities are different in the two images, the resulting image shows such regions with green or magenta colors. This method is the default one while using the fusing function if no other method is specified.
- 'blend': This method simply overlays two images using 'alpha blending'. In alpha blending, a translucent foreground color is combined with a background color in order to produce a new color that is blended between the two. The point to note here is that the translucency of the foreground color can range from completely opaque to completely transparent.
- 'checkerboard': As the name itself indicates, the output image will have alternating rectangular regions from the two input images.
- 'diff': This method simply creates the difference images of the two given input images.

- ‘montage’: With this method, the two images are kept next to each other (side to side) forming one single image.

The MATLAB function and the method used for fusing our images will be discussed in the following chapter. But how the fused image looks like when our spectrogram and mel spectrogram are combined is shown in figure 7.

3.6 Sampling Rate

In signal processing, the reduction of continuous time-signal to a discrete time-signal is called sampling. That means, a continuous signal will be sampled or divided into smaller parts to form a discrete time signal. In addition, as we all know, rate is nothing but the varying parameter per second. Thus, sampling rate in signal processing is defined as the number of samples of audio carried per second. It is measured in Hertz (Hz). For example, if a signal is said to have a sampling rate of 42000 Hz or 42kHz, that means it has 42000 samples of audio per second.

3.7 MFCCs

Mel Frequency Cepstral Coefficients (MFCCs) is a feature that is being widely used in sound classification, speech, and speaker recognition for a very long time now. Since 1980’s when they were introduced, they have been state-of-the-art feature in the audio field. MFCCs are the cepstral coefficients that are in mel scale, as explained in the previous section.

But what does ‘cepstral’ mean? Cepstrum is the information of rate of change in spectral bands. To understand this in detail, we know that frequency domain is obtained when we apply Fourier transform on the time signal. Now, if we take log of the magnitude of this Fourier spectrum and then again take the spectrum of this log by applying the cosine transform, then the resulting spectrum is called cepstrum. In other words, cepstrum is obtained by performing cosine transformation on the log of the frequency spectrum itself.

MFCCs are the coefficients that collectively make up a Mel Frequency Cepstrum (MFC). To understand this, we need to first understand the process of how we derive these coefficients mathematically. The steps are as follows:

- Take Fourier transform of a windowed signal
- The powers of the spectrum obtained above are mapped onto the mel scale.

- Logs of the powers at each of the mel frequencies are taken.
- Assuming the mel log powers as signals, apply Discrete Cosine Transform (DCT).
- The amplitudes of the resulting spectrum (here, cepstrum) are the MFCCs.

To sum up the above steps, if we apply DCT to spectrograms, it gives nothing but the MFCCs. Because we learnt that spectrogram is the frequency domain representation of the given time signal in log scale after all. Also, after extracting these MFCCs, many researchers use 2^{nd} to 13^{th} coefficients to train their neural networks and discard the rest.

3.8 Convolutional Neural Networks

Most of the deep learning tasks are based on Convolutional Neural Networks, which are also called as ConvNets. In CNNs, there are input layer, an output layer and multiple hidden layers, among which at least one of the hidden layers is 2D convolutional layer, that applies a mathematical linear operation, convolution, to convolve learned features with input data, thus making it well suited for analyzing pixel data like 2D images and hence the name, Convolutional NNs. CNNs do not need manual feature extraction, they simply learn from the patterns of the training images, which makes deep learning desirable to work on areas like object classification. CNN requires much lower pre-processing when compared to other classification algorithms. Also, unlike regular neural networks, CNN's neurons in one layer are not connected to all the neurons of the next layer, and layers are 3-dimensional with height, width and depth.

- Convolutional Layer: Convolutional layer is the core layer of a CNN that does most part of the computational work. The mathematical operation, convolution takes place in this layer. For regular neural networks, the input is a vector whereas, CNNs take multi-channeled input images. In other words, CNNs operate over volumes. To understand what this layer does exactly, let us start with saying, its parameters consist of a set of learnable filters (or kernels). During forward pass, we convolve (or slide) each filter across the width and height of the input volume, and at any position, compute the dot product of the input and the entries of the filter. The filters thus produce activation maps when they see any visual features in the input. Stacking up each activation map generated by each filter, produces the output volume.

Three hyperparameters – depth, stride and zero-padding control the size of the output volume. Depth controls the number of neurons we would like to allocate to look for something different in the input. Stride specifies the size of our sliding window. If stride is 1, then we move the filter one pixel at a time. Zero-padding allows us to control the spatial size of the output volumes, by adding (or padding) zeros to the input volume.

- Rectified Linear Unit (ReLU) Layer: An activation function is responsible for transforming the weighted input into output. Rectified linear unit applies a piecewise linear activation function that outputs the input directly if it is positive, otherwise, simply outputs zero, which is why the rectifier function is defined as the positive part of its argument. This activation function is widely used in many neural networks because of its less complicated math and better performance. This function overcomes the vanishing gradient problem and converges faster, which makes it a default activation for CNNs.
- Pooling Layer: Pooling layer is used to reduce the spatial size of the convolved feature which helps in decreasing the computational power required to process the data. We can also say that it reduces overfitting. Pooling layer is used between convolution layers and by extracting only dominant features, and neglecting other activations received from the initial convolutional layers, it forces the next convolution layers to learn from the limited data provided from the activations.

  There are two types of pooling – Max Pooling and Average Pooling. In max pooling, maximum value is returned from the area of the image covered by the kernel or filter, whereas in average pooling, average value of all the values from the image portion covered by the kernel is returned. Max pooling is the most used approach because it also performs noise suppression along with dimensionality reduction.

- Fully connected (FC) Layer: In a fully connected layer, as the name itself says, all its neurons are connected. Neurons in FC layer are like those in regular neural networks and work in a similar manner, as they have connections to all the activations of the previous layer.

3.9 Recurrent Neural Networks (RNNs)

In a recurrent neural network, information persists and thereby, they are the networks with loops. RNNs process sequential data using their internal memory. While training, RNNs remember things learned from previous inputs. They store memory and use this memory from the so-called hidden state vector and apply that on current inputs to generate outputs. So, a different output can be produced by the same input depending on the previous inputs in the sequence or series. RNNs are widely used in speech recognition, handwriting recognition etc.

- Long Short-Term Memory (LSTM) Network: In the field of deep learning, LSTM [59] is an artificial recurrent neural network architecture. Regular neural networks usually have feedforward connections. On the other hand, LSTM networks have feedback connections. LSTMs are mostly RNNs except for an addition of little more mathematics. LSTMs have gated cells, which will have information outside the normal flow of the recurrent network. These gated cells learn what to allow and what to discard by iterations of making guesses, backpropagation of error, and adjusting weights using gradient descent.

3.10 Pre-trained Deep Neural Networks

In this thesis, several pre-trained deep networks that are available online are used. They are called 'pre-trained' because they are already trained and tested with a large dataset of 1000 classes for image classification. In this section, we will see about each network that has been used at any stage in this thesis in detail.

3.10.1 AlexNet

The authors Krizhevsky et al., [47] has proposed a deep convolutional neural network to participate in the competition of ImageNet Large-Scale Visual Recognition Challenge (ILSVRC) 2010. The purpose of the competition is that the network should be able to classify images from the ImageNet database [60], which contains 1.2 million high-resolution images that belong to 1000 classes. The results of this network in the year 2010 were better than the previous state-of-the-art. In ILSVRC-2012, they entered a variant of this model again and won the contest.

The network is 8 layers deep, which comprises of five convolutional layers and three fully connected layers. Some of these convolutional layers are followed by max-pooling layers. Also,

after the last fully connected layer, there is a SoftMax layer of 1000-way, followed by a classification output layer. The SoftMax layer is 1000-way because the ImageNet database used contains images from 1000 classes and the SoftMax layer recognizes the class of the input based on the features extracted from the previous layers in the network. Classification Output is responsible to show the output given by the SoftMax layer. In fully connected layers, the "dropout" regularization method is used in order to reduce overfitting. The first layer, data input layer accepts images of size 224*224 pixels.

3.10.2 GoogleNet

Szegedy et al., [48] proposed a deep convolutional neural network for classification and recognition of images to participate in the ILSVRC-2014 challenge. The network stood as the winner that year by classifying images into 1000 classes from ImageNet database.

This is a very deep network having a total of 100 layers, with 22 layers of depth. Two new things called Inception layers and embeddings were employed to the layers. The inception layers are responsible to perform local sparse abstractions of the input and the embeddings are like pooling layer in terms of functionality. This network also has SoftMax layer and Classification output layer at the end, and an image input layer at the beginning. The input layer accepts images of size 227*227 pixels and the SoftMax used is a 1000-way layer.

3.10.3 ResNet

Kaiming He et al., [61] proposed a deep convolutional neural network called ResNet, which stands for Residual Network. This network was developed to participate in the contest of ILSVRC-2015 for task of classification of images. The result has won the 1st place on the classification task that year.

The intention behind presenting the ResNet is to ease the training of deeper networks by explicitly reformulating the layers as learning residual functions with reference to the input layers, instead of learning unreferenced functions. This network was able to classify more than a million images into 1000 different classes from the ImageNet database. It has an image input size of 224*224. This residual network has more than one version – ResNet-18, ResNet-50, ResNet-101 with 18, 50 and 101 deep layers, respectively.

3.10.4 VGG

Simonyan et al., [62] developed a deep convolutional neural network called VGG network. VGG stands for Visual Geometry Group from the University of Oxford. This network had participated in ILSVRC-2014 ImageNet Classification contest and secured the first and second places in localization and classification tasks, respectively.

By contributing to the evaluation of networks of increasing depth, the VGG group has shown that there is significant improvement in the performance than AlexNet by increasing the depth to 16-19 layers. This network has used 3*3 filters in all the convolutional layers to reduce the number of parameters in deep networks. VGG has two models – VGG 16 and VGG 19 which have 16 and 19 weight layers of depth, respectively. We used VGG 16 in this thesis and the network has an image input layer with 224*224 in size.

3.10.5 ShuffleNet

Zhang et al., [63] introduced a deep CNN architecture called the ShuffleNet to participate in ILSVRC-2017 competition. This network had achieved 89.8% in Top 5 Accuracy metric value by classifying images that belong to 1000 classes from ImageNet database.

This was built as a computation-efficient network, especially for mobile devices with very limited computational power. In this architecture, two new operations – pointwise group convolution and channel shuffle are used, which helps in reducing computation cost but maintaining the accuracy. The operation, channel shuffle aids in building more powerful structures with multiple group convolutional layers. The network's input layer accepts the images of 224*224 pixels.

3.10.6 SqueezeNet

Landola et al., [64] proposed a small yet efficient deep neural network named as SqueezeNet. This network is just 18 layers deep unlike many other popular deep networks with lot of layers. The network achieved AlexNet-level accuracy only with 18 deep layers.

The purpose of this network is to have a small deep network with a smaller number of layers and be efficient at the same time in terms of accuracy. There are a few advantages of smaller DNNs like they require less communication between servers during training, requires less

bandwidth to export the model from the cloud, more feasible to deploy on hardware requiring less memory, faster in computing etc. Thus, SqueezeNet has all these advantages and performs as well as AlexNet with 50x fewer parameters. This network was trained on ImageNet database and can classify images into 1000 classes. The image input size of this network is 227*227.

3.10.7 Inception V3

Szegedy et al., [48] proposed a CNN called Inception V3. The aim of this network is to utilize the computational task as efficiently as possible. This network has an increased model size with many deep layers and thus high computational cost. But it is as efficient and uses a smaller number of parameters when compared to other state-of-the-art networks.

This network has participated in the ILSVRC 2012 ImageNet Classification challenge and yielded great results. Inception V3 is 48 layers deep and takes longer to compute. The network can classify more than a million images from the ImageNet database into 1000 classes. The image input size of this network is 299*299.

3.11 Machine Learning Classifiers

A classifier performs the task of classification, which is the prediction of class of given data points. There can be two or more classes. Training data will be utilized by a classifier to understand how a given data point is related to a class, and then to classify new or test data, which is the reason why classification is considered to be a category of supervised learning. There are several classifiers in Machine Learning – Naive Bayes, Support Vector Machines (SVMs), Nearest Neighbor, Decision Trees, Random Forest. Each of these classifiers follow their own principles and classify data. We cannot say beforehand which classifier is best for a given task. So, it is better to train all the classifiers and see which of them gives best results.

- *Support Vector Machines:* An SVM [65] is a supervised machine learning model. It is inherently a binary classification model, that classifies new inputs to one or the other of the two categories when trained with a set of training examples. If there are two classes of data points, then there are many possible hyperplanes that can be chosen to separate them. But the objective of the SVM algorithm is to choose a hyperplane that distinctly classifies the data points with maximum margin. SVMs perform transformations to the given data using a technique called the kernel trick.

- *K-Nearest Neighbor (KNN):* KNN [66] is one of the simplest machine learning algorithms. It is a non-parametric method used for both classification and regression. It functions based on feature similarity and distance algorithm. The model simply calculates the distance between the given data point to other data points based on features similarities and maps to the closest data point.

3.12 MATLAB

"MATLAB R2019b, The MathWorks, Natick, 2019" [67] is the software developed by MathWorks that was used in this work. MATLAB has many toolboxes and in-built functions that makes the life of a machine learning practitioner easier. Its documentation has everything to help any user with how to utilize its functions and proceed with programming.

- *Deep Learning Toolbox and Audio Toolbox:* Deep Network Designer toolbox and Audio toolbox of MATLAB are used in this work to train deep networks and deal with audio files, respectively. Deep Learning toolbox is a user-friendly space where even a beginner can train deep networks easily. There are different layers available in this toolbox that we can use in modifying a pretrained network or building our own network. Audio Toolbox has all the functions that can be applied to audio files. Many in-built functions like spectrogram, melSpectrogram etc., are used to generate image representations of the audio files.
- *Classification Learner App:* In MATLAB, the Statistics and Machine Learning Toolbox has classification learner app that consists of all the classifiers. User just needs to give the input data and can train all the classifiers or any number of desired classifiers very easily. We can either enable or disable the usage of cross-validation property in the app. Accuracies of all the classifiers can be seen right beside each of them and the results can also be viewed in the form of confusion matrix or ROC curve (Receiver Operating Characteristic curve) etc.

4 Experiments and Results

In this chapter, different experiments that are conducted and the results obtains will be presented. The chapter is mainly divided into two sections. The first section describes the audio dataset that has been used in this work whereas, the second section discusses the different experiments conducted and the results obtained through the experiments. The sections are sub-divided as per convenience. The following figure 8. gives the details of the workflow of the performance analysis.

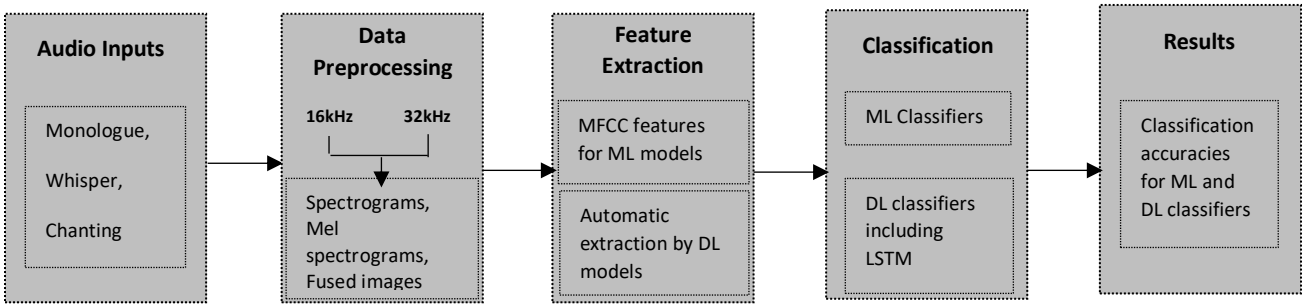


Figure 8. Workflow

4.1 Audio Dataset description

Through this research, our intention is to contribute to audio classification by successfully classifying various styles of people talking. For this experiment, we chose 3 classes – chanting, whispering, monologue. The reason behind selecting these categories of talking styles is that they are distinguishable based on the range of pitch while talking in that corresponding style. The voice or tone of the person can make us understand what the talking style is. For example, when a human listen to a whispering audio and a normal monologue audio, he can easily distinguish between both of them. This is because of the way the pitch will be for each style of talking. Hence, our work is to make sure a computer algorithm can recognize and should be able to classify the type of talking style, just like how a person does. Since this work just began to apply deep learning models to speech mode classification, we wanted to start off with most distinguishable classes and see how the deep networks work. Later in future, this can be taken further by adding more classes which can be close to each other and a bit challenging to recognize and classify.

The three classes selected are chanting, whispering and monologue. The audio clips of each of these classes are collected from YouTube. Each class has 14 recordings of slightly varying durations (01:58-02:03 minutes). There is no more than one recording per each subject, which makes it 14 subjects for each class. So, all these videos from YouTube are initially converted to .wav files using online tools, and then trimmed to the specified durations just so that they are uniform and there wouldn't be much difference in the number while generating spectrograms and mel spectrograms for each subject.

- ***Chanting:*** These 14 audio clips are mostly of Vedic mantras, Tibetan chants, Indian mantras in temple etc. Quite a few of them are also mixed with a background music along with the chants. A few of our recordings are of male voice, some are of female voice, and the rest are mixed and a group of people singing. This class stands out because of the music while chanting and also a group of people doing that at the same time, which is why its graphical representation would look different from other classes.

- ***Whispering:*** This class also has 14 recordings, one each of 14 different people. There are male and female recordings among the fourteen clips. All of these clips are ASMR videos available in YouTube. ASMR stands for Autonomous Sensory Meridian Response, and describes a euphoric tingling feeling when a person hears or watches certain sounds. To create this therapeutic feeling, these videos are made without any loud noise or talking, but just whispering and such soothing sounds. Since there is no noise or loud talking involved, the pitch of whispering sounds seems low and generates spectrograms with lower amplitudes when compared to other classes.

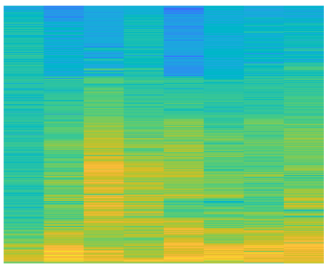
- ***Monologue:*** Monologue (or simply, talking) is the most common style in general, for people while communicating. This class also has 14 audio clips of around two-minute length from 14 subjects, with male and females talking. There are plenty of monologue videos available on YouTube and we collected using online tools. This class is not similar to chanting or whispering, thus, has different graphs generated in terms of amplitudes.

4.1.1 Generating 2D images

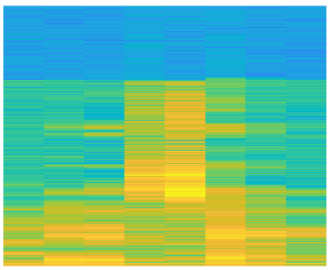
The above collected audio samples have to be converted to their corresponding image representations so that those images can be used to train the pre-trained deep networks that use images as their inputs to classify them into different classes. The three 2D image representations that are chosen in this thesis are spectrograms, mel spectrograms, and fused images by combining the spectrograms and mel spectrograms. In order to generate these images, each audio clip was initially split into 2 sec audio blocks and conversion to spectrogram or mel spectrogram is then applied to each block. This means that the entire audio file is divided into small clips of window size 2 secs and 50% overlap is applied. The number of audio blocks per each audio file is based on dividing the whole file into 2 sec periods going in steps of the sampling rate used with 50% overlap, until the end of the entire file length is reached.

- Spectrogram: To convert audio to spectrograms, the MATLAB function "spectrogram(x, fs)", where 'x' is the input signal and 'fs' is the sampling frequency has been used in this thesis. This function uses Short-time Fourier Transform (STFT) in order to generate the spectrograms.
- Mel spectrogram: To convert audio to mel spectrograms, the MATLAB function "melSpectrogram(x, fs)", where 'x' is the input signal and 'fs' is the sampling frequency has been used in this thesis. This function is based on the linear cosine transform of a log power spectrum on a nonlinear mel scale of frequency.
- Fused image: In this step, we intend to superimpose spectrograms with their corresponding mel spectrograms, and this superimposing is referred to as 'fusing' in this work. The MATLAB function, 'imfuse(A, B, method)' has been used to fuse the images i.e., generate the composite of two given images. In the function, A and B are the two input images that are supposed to be fused and 'method' is the type of fusion we want to specify. The methods available in MATLAB are discussed in Chapter 3. In this work, we used 'blend' method for fusing our A – spectrogram with its corresponding B – mel spectrogram.

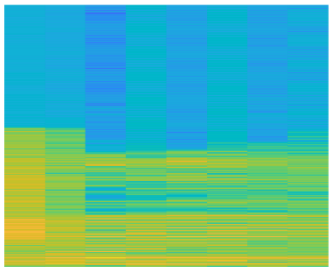
In this work, we are focused on studying the effect of sampling rate on the classification accuracies. Thus, instead of using the default sample rates for the audio samples, we considered two sampling rates – 32kHz and 16kHz. We generated two sets of corresponding spectrograms, mel spectrograms and fused images – one set for each sampling rate. The number of images generated varies with



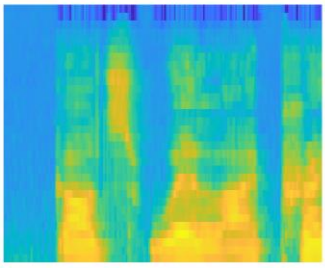
Spectrogram - Monologue



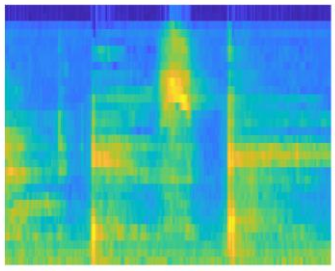
Spectrogram - Whispering



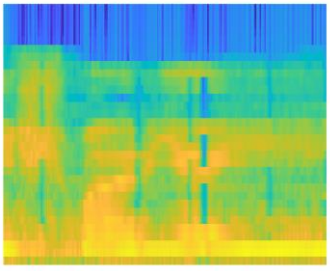
Spectrogram - Chanting



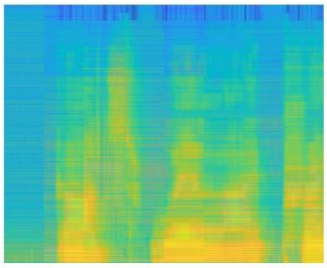
Mel Spectrogram - Monologue



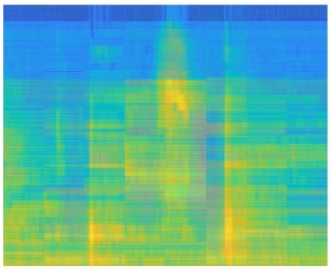
Mel Spectrogram - Whispering



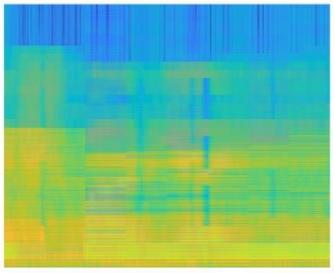
Mel Spectrogram - Chanting



Fused - Monologue



Fused - Whispering



Fused - Chanting

Figure 9. Example spectrograms, mel spectrograms, fused images of monologue, whispering and chanting classes

each sampling rate. Table 1 summarizes the information about the number of images per class for the given sampling rate. Also, how the spectrograms, mel spectrograms and fused images look like are shown in figure 9.

4.2 Preliminary Experiments

Initially, preliminary experiments were conducted to see the performance of various machine learning classifiers and pre-trained deep networks. Through these experiments, we could estimate the performance of each technique/network by studying the balance between computational cost and classification accuracies. In this preliminary stage, we studied the basic performance of all the ML classifiers available in Classification Learner App of MATLAB, and all the pre-trained deep

Table 1. Number of images available per category in this dataset

Class	Number of images generated per category (spec, mel spec, fused) for each sampling rate	
	32KHz	***16KHz***
Monologue	2406	4801
Whisper	2372	4734
Chanting	2548	5095

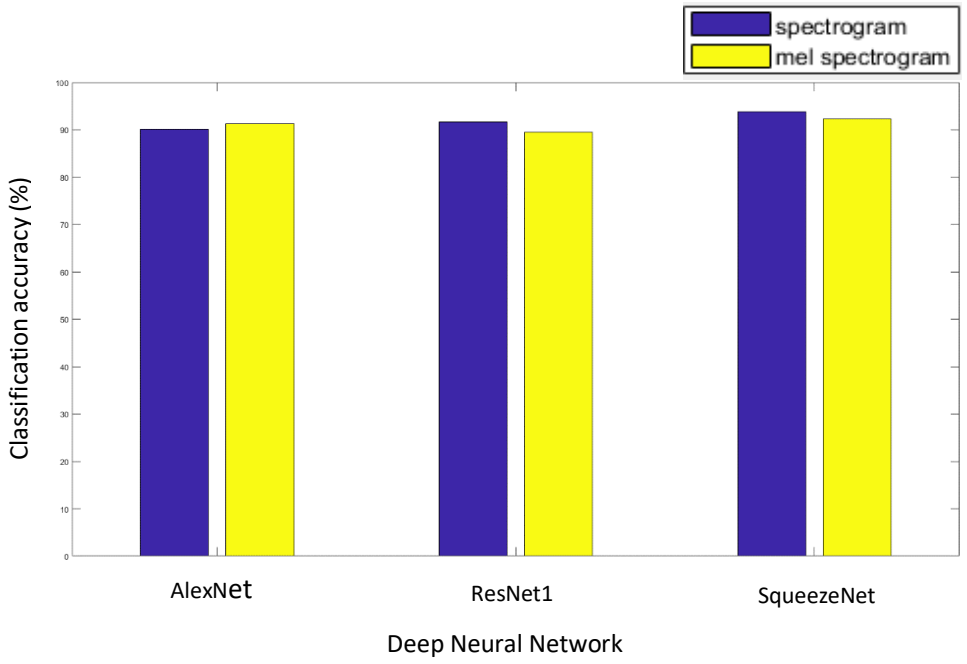
networks that were mentioned in Chapter 3. For this study, only 500 images of each class were used. With the results obtained, many of the classifiers were eliminated due to the longer time taken for computation and also unsatisfactory accuracies. Most of the eliminated classifiers yielded less than 50% accuracies and took more than 1 hour of computational time, considering the fact that only 500 images of each class were given as the input. Thus, main experiments were conducted with a greater number of images and with classifiers that yielded basic accuracies of at least 60% in the preliminary stage.

4.3 Experiments

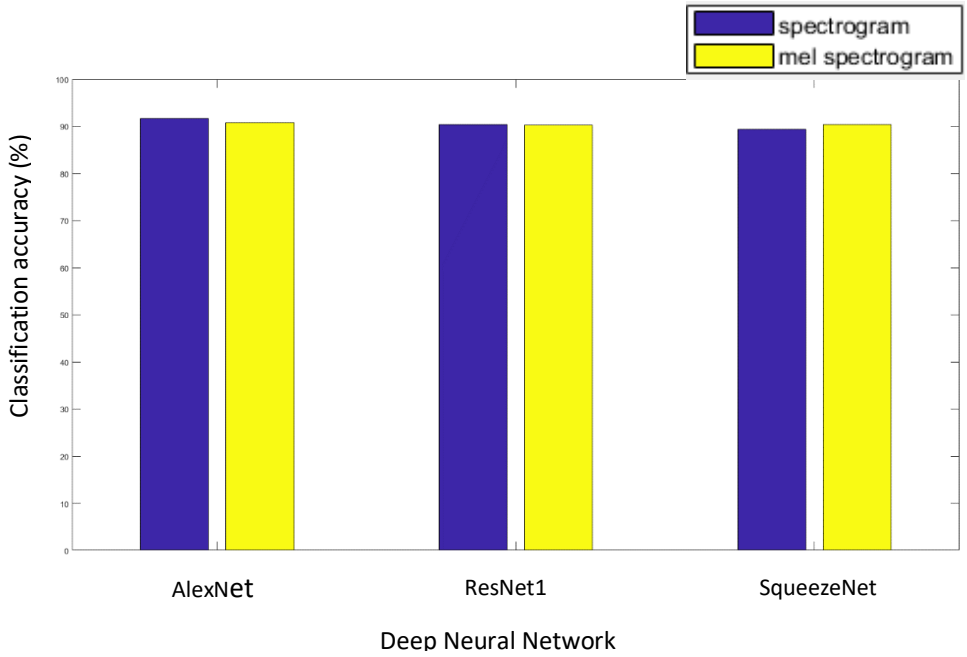
The experiments that are conducted in this work are divided into four sub-sections. Since it is a comparison study, classification results of machine learning based algorithms are compared to deep learning algorithms in different aspects. Thus, this section includes the details about the four sets of experiments in each sub-section and the results obtained. Here is the list of experiments we have performed to classify three audio classes, namely, (1) chanting, (2) monologue and (3) whispering, as specified in the previous section.

- Using ML and DL classifiers, including SVM, KNN, and three pre-trained deep networks.
- Fusion (score and feature level) of ML Classifiers
- Fusion (score and feature level) of DL Classifiers
- Fusion (image level) of spectrograms and mel-spectrograms, supported by DL algorithms.

- Using Long Short-Term Memory Networks (LSTMs) to process the extracted features from each deep network at 32kHz i.e., our LSTM is supported by our pre-trained deep networks in our



a) 32kHz sampling rate



b) 16kHz sampling rate

Figure 10. Classification accuracies of three deep neural networks at 32kHz and 16kHz sampling rates

study at the most efficient sampling rate, which is 32kHz.

What follows is a deeper discussion on the aforementioned experiments.

4.3.1 Classification using ML and DL classifiers

In this set, we performed the basic classification technique by selecting two machine learning classifiers and three deep learning classifiers as shown in Figure 11.

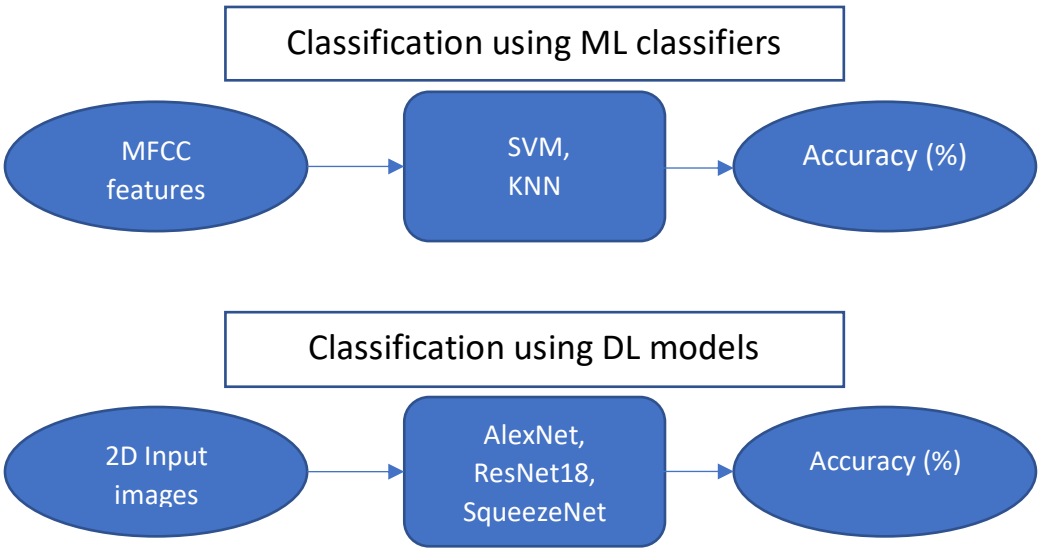


Figure 11. Classification using ML and DL classifiers

In order to perform classification using machine learning classifiers, feature extraction needs to be done first. So, we chose to extract the state-of-the-art feature, Mel-Frequency Cepstral Coefficients (MFCCs) from the audio samples. These MFCCs have proven to be giving great classification results in many audio classifications studies. Thus, from the available audio files, MFCC features are extracted first and these extracted features are saved to a .mat file (MATLAB). We train two of the selected ML classifiers – Support Vector Machines (SVM) and K-Nearest Neighbor (KNN) with the saved MFCC features. The results obtained from these classifiers are shown in Table 2.

Table 2. Classification accuracies for pre-trained deep networks and machine learning classifiers

Pre-trained network/Classifier	**Classification accuracies (%)**			
	32kHz		***16kHz***	
	Spect	***Mel spect***	***Spect***	***Mel spect***
AlexNet	90.1	91.3	**91.7**	**90.8**
ResNet18	91.7	89.5	90.4	90.3
SqueezeNet	**93.8**	**92.3**	89.4	90.4
SVM Classifier	79.7			
KNN Classifier	68.3			

Three pre-trained deep neural networks – AlexNet, SqueezeNet and ResNet18 are chosen to classify our three speech mode classes based on two sampling rates. As all these networks were pretrained with ImageNet dataset, we performed transfer learning by replacing a couple final layers like classification output layer, fully-connected layer etc., and also changing the parameters using Deep Network Designer app of MATLAB, in order to obtain the best results possible. The dataset is split into 1650 training images and the rest for testing, and 3400 images in training set and the rest to be used as the test set in the cases of 32kHz and 16kHz sampling rates respectively. All the three networks are trained separately with spectrograms and mel spectrograms for each sampling rate, and all the classification accuracies were noted. The results are summarized in Table II and it shows that AlexNet and SqueezeNet yield the highest accuracies for 16kHz and 32kHz sampling rates, respectively. Overall, SqueezeNet yielded the highest accuracy of 93% in the case of 32kHz spectrograms. A graphical representation of the classification accuracies of the pre-trained deep networks for both sampling rates are shown as bar graphs in Fig. 10. Note that the computational speed of AlexNet and SqueezeNet is twice as much as that of ResNet18.

4.3.2 Fusing ML and DL based Classifiers

In this stage, we combined machine learning classifiers with deep learning classifiers to determine if there will be any significant improvement in terms of classification accuracies. To conduct such a study, we fused the ML and DL classifiers at score-level and feature-level as shown below in Figure 12. In the case of score-level fusion, we simply fused each ML classifier (SVM and

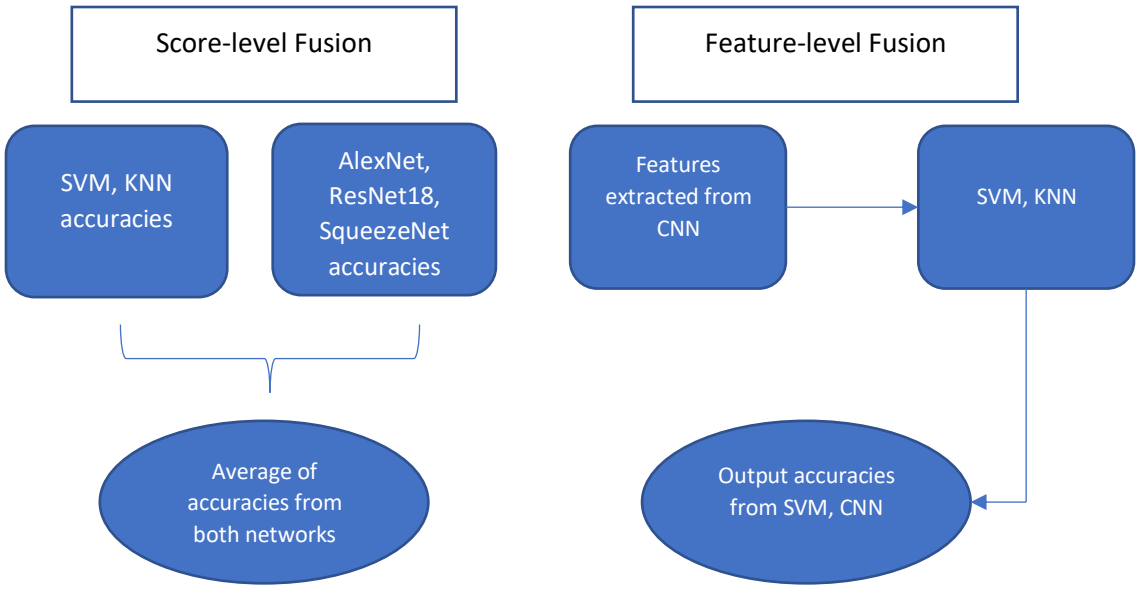


Figure 12. Fusing ML and DL based Classifiers

KNN) with each DL classifier separately, by taking the average of the accuracies that were presented in Table 2.

In the case of feature-level fusion, we extracted the features that were learned and generated by the deep learning classifiers themselves, and these extracted features are used as input to the machine learning classifiers. For instance, we initially trained AlexNet with our 2D images (e.g. Spectrograms) and extracted features from the last fully connected layer of the AlexNet, that were generated because of that training. The reason behind selecting the last fully connected layer of all is because the given input images will pass through the initial layers of the deep neural network and only the most prominent features possible will be at the final layers. So, we extracted the features from the final deep layer of each network and gave them as an input to SVM and KNN for classification. The comparison of the results of score-level fusion and feature-level fusion is shown in Table 3. It is evident from the results that feature-level fusion gave better results than those of score-level fusion by around 5%-15%.

Table 3. Accuracies obtained by combining ML and DL based classifiers

Method	32kHz				16kHz			
	Score-level fusion accuracy (%)		*Feature-level fusion accuracy (%)*		*Score-level fusion accuracy (%)*		*Feature-level fusion accuracy (%)*	
	Spect	*Mel spec*	*Spect*	*Mel spec*	*Spect*	*Mel spec*	*Spect*	*Mel spec*
Alex+SVM	84.9	85.5	91.6	**91.8**	**85.7**	**85.3**	91.2	90.8
Alex+KNN	79.2	79.8	91.4	90.8	80.0	79.6	91.4	90.7
ResNet18+SVM	85.7	84.6	92.0	89.5	85.0	81.7	**91.8**	90.3
ResNet18+KNN	80.0	78.9	91.9	89.9	79.3	75.0	91.5	90.5
SqueezeNet+SVM	**86.8**	**86.0**	***92.1***	90.8	84.6	85.0	91.5	**91.6**
SqueezeNet+KNN	81.1	80.3	91.2	90.3	78.9	79.4	89.7	89.2

4.3.3 Fusing Spectrograms and Mel spectrograms

Unlike in the previous section, where we combined machine learning and deep learning, in this set of experiments, we fused spectrograms and mel spectrograms at two levels in order to

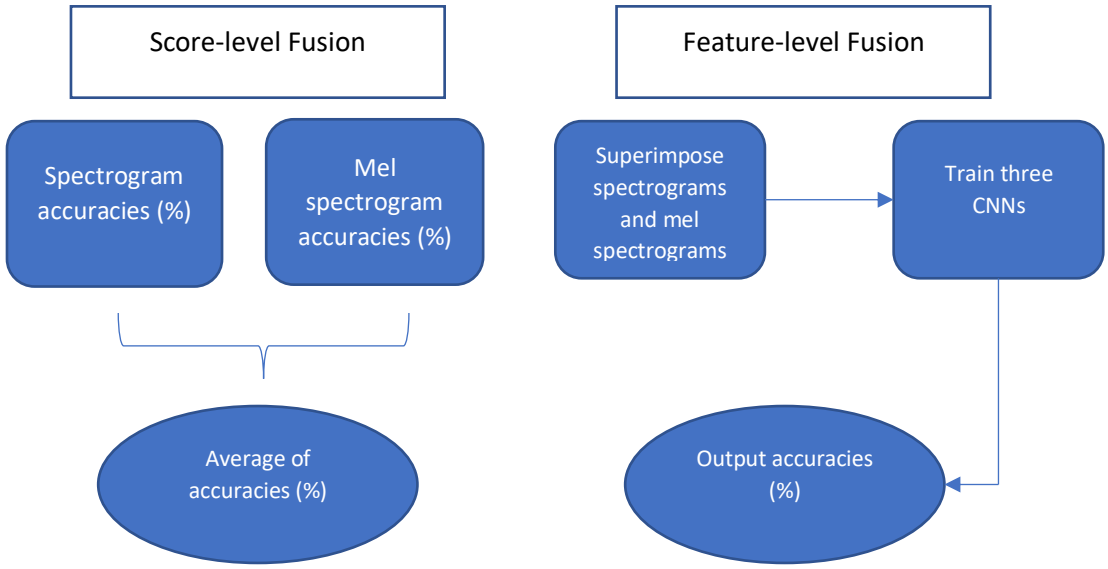


Figure 13. Fusion of Spectrograms & Mel Spectrograms

compare as shown in Figure 13. One is score-level fusion and the other is image-level fusion. Similar to the score-level fusion in Section III.B, we combined the accuracies of those obtained for spectrograms and mel spectrograms and calculated the average. This is repeated for all the three deep networks used.

For image-level fusion, fused images are generated by superimposing spectrograms and mel spectrograms as mentioned in the previous section. Examples of these fused images for both monologue and whispering classes is shown in the Fig. 9. Now, these new images are used as the inputs to our deep networks and the classification rates are noted and compared to those of score-level fusion. The results of this experiment are shown in Table 4. We can say from the table that fusing spectrograms and mel spectrograms does not improve performance. SqueezeNet at 32kHz sampling rate achieved 92.5% of accuracy with score-level fusion.

Table 4. Accuracies obtained by combining spectrograms and mel spectrograms

Method	**32kHz**		**16kHz**	
	Score-level fusion	***Image-level fusion***	***Score-level fusion***	***Image-level fusion***
AlexNet	90.7	87.7	89.5	88.1
ResNet18	90.7	89.8	**90.9**	88.6
SqueezeNet	**92.5**	87.2	89.1	88.9

4.3.4 Classification using LSTMs (Our proposed method)

We built a basic LSTM network in MATLAB to run this set of experiments as shown in Figure 14. Our network has a 'bidirectional LSTM (BiLSTM) layer' with 50 hidden units. To implement this, the MATLAB function "layer = bilstmLayer(numHiddenUnits, Name, Value)" has been utilized.

Figure 14. Proposed Method

We decided to combine an LSTM network with the deep CNNs used in this work and study the classification accuracy of our LSTM network at 32kHz. In detail, the generated features of AlexNet, ResNet18 and SqueezeNet are extracted and these extracted features are used to train our LSTM network. Thus, in a way, both the DNN and LSTM now have the same features but the only difference is that, the LSTM network will have one more fully-connected layer to generate new features from these given features, unlike the DNN, whose features are already finalized. In the case of earlier sub-section where we combine ML and DL classifiers, the network does not get any deeper. So, this set of experiments was expected to yield better performance results. Hence, the

Table 5. Accuracies obtained from our proposed LSTM-CNN network

Method	***Spectrograms***	***Mel spectrograms***	***Fused***
AlexNet + LSTM (32kHz)	91.9	90.3	88.6
ResNet18 + LSTM (32kHz)	90.7	91.2	87.1
SqueezeNet + LSTM (32kHz)	***95.7***	90.4	88.9

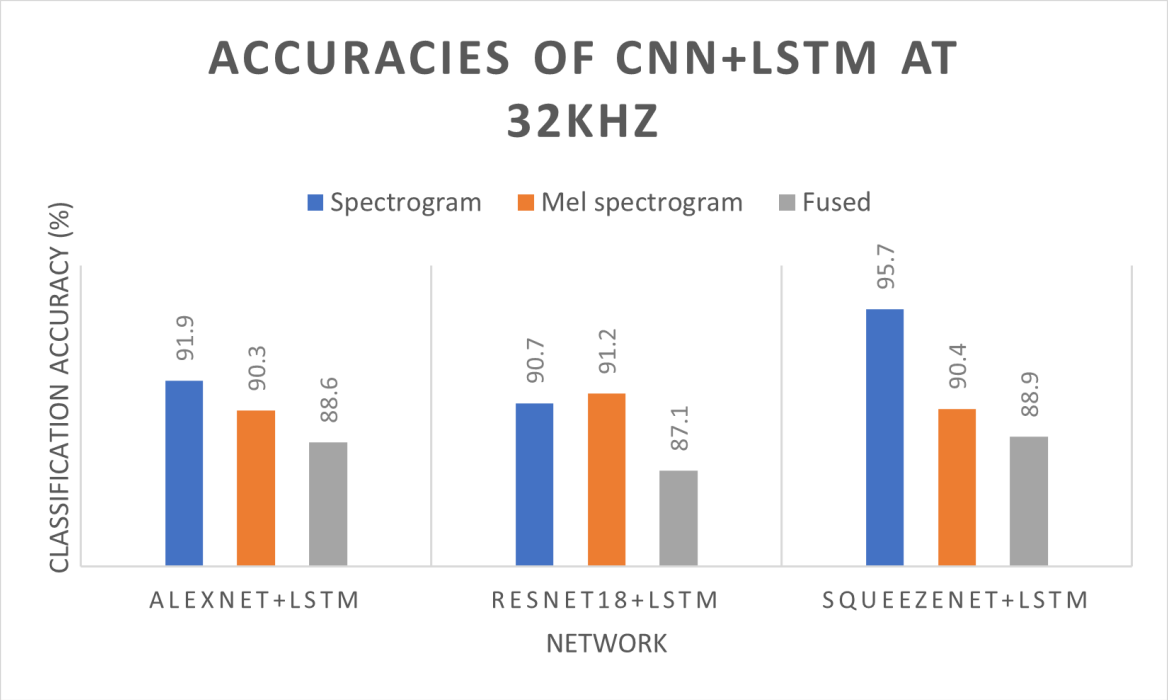


Figure 15. Results Overview of Proposed Method

results of LSTM further improved the accuracy by around 3.2% in the case of spectrograms at 32kHz when compared to the SqueezeNet alone. The LSTM accuracies of in the case of other two networks – AlexNet and ResNet18 are also improved. The results obtained with LSTMs are shown in Table 5, and a graphical representation of the results obtained with the proposed method is shown in Figure 15.

5 Conclusions and Future work

In this thesis, the challenge to solve a three-class speech mode classification problem – monologue, whisper, and chanting, using machine learning and deep learning classifiers has been taken up. In this comparison study, we performed experiments with four stages. In the first stage, we compared ML and DL classifiers, and clearly the deep learning methods that used spectrograms and mel spectrograms as the inputs outperformed ML classifiers, which used MFCC features. Of all the three deep networks tested, SqueezeNet yields the highest classification rates i.e. 93% and 91.9% at 32kHz for spectrograms and mel spectrograms, respectively. On the other hand, AlexNet seems to yield superior accuracy when using the 16kHz sampling rate.

The results from the second stage of experiments involving the fusion of DL and ML classifiers at score-level and feature-level prove that the feature-level fusing gives much better results when compared to score-level fusion, by improving the accuracy from around 5% to 15% depending on the combination used.

The third stage of experiments were used to compare the results of score-level and image-level fusions of spectrograms and mel spectrograms. Although, there is no big difference observed in the accuracies in this experiment, score-level fusion gave slightly better results than the fused images. With this study, we can say that SqueezeNet among the deep networks, yield the highest classification accuracy, i.e. more than 91% with both spectrograms and mel spectrograms, when combined with SVM and KNN, using feature-level fusion and when sampling at a rate of 32kHz.

Moreover, in the fourth stage of experiments, with our proposed method of LSTM-CNN, there is a further improvement in classification accuracies by 3.2% with spectrograms at 32kHz using LSTM network when compared to SqueezeNet alone. Thus, the LSTM network yielded an average accuracy of 95.7%. Overall, from the results of all stages of experiments it is clearly evident that the 32kHz sampling rate gave slightly better results by at least 2% than those of 16kHz.

Since the topic, speech mode classification itself is not being researched that well in the field of audio classification, there is a lot to explore in this area. With regards to the work done in this thesis, in addition to the study of the effects of sampling rates, more factors and their effects on classification accuracies can be studied in our future work. The study of sampling rate can be

further extended by letting the audio samples to have their default sample rate and see how that would affect the results. Other factors like noise can be added to the audio files and add denoising methods to have real-time applications. The usage of classification techniques can be extended to a much wider range and varieties of combinations can be tried. Application of LSTMs has great scope and can be widely explored in this area. Most importantly, this three-class classification problem can be made a multi-class speech mode classification problem by adding several other classes like screaming, shouting, crying etc., to have a more challenging task.

6 Bibliography

[1] B. Raphael, "Artificial Intelligence," *IEEE,* vol. 6, no. 5, pp. 9-10, May 1973.

[2] P. Louridas and C. Ebert, "Machine Learning," *IEEE,* vol. 33, no. 5, pp. 110-115, Aug 2016.

[3] N. Jmour, S. Jayen and A. Abdelkrim, "Convolutional Neural Networks for Image Classification," *IEEE,* pp. 397-402, June 2018.

[4] C. A. U. Hassan, M. S. Khan and M. A. Shah, "Comparison of Machine Learning Algorithms in Data Classification," *IEEE,* pp. 1-6, 2018.

[5] J. Patterson and A. Gibson, Deep Learning: A Practitioner's Approach, Beijing: O'Reilly, 2017.

[6] U. Zolzer, Digital Audio Signal Processing, Wiley, Aug 2008.

[7] N. Davis and K. Suresh, "Environmental Sound Classification Using Deep Convolutional Neural Networks and Data Augmentation," *IEEE,* pp. 41-45, 2018.

[8] H. Zhou, Y. Song and H. Shu, "Using deep convolutional neural network to classify urban sounds," *TENCON 2017 - 2017 IEEE Region 10 Conference, Penang,* pp. 3089-3092, 2017.

[9] G. Tzanetakis and P. Cook, "Musical genre classification of audio signals," *IEEE Transactions on Speech and Audio Processing,* vol. 10, no. 5, pp. 293-302, July 2002.

[10] S. P. Todkar, S. S. Babar, R. U. Ambike, P. B. Suryakar and J. R. Prasad, "Speaker Recognition Techniques: A Review," *2018 3rd International Conference for Convergence in Technology (I2CT), Pune,* pp. 1-5, 2018.

[11] F. Beritelli and R. Grasso, "A pattern recognition system for environmental sound classification based on MFCCs and neural networks," *2008 2nd International Conference on Signal Processing and Communication Systems, Gold Coast, QLD,* pp. 1-4, 2008.

[12] T. Kim, J. Lee and J. Nam, "Comparison and Analysis of SampleCNN Architectures for Audio Classification," *IEEE Journal of Selected Topics in Signal Processing,* vol. 13, no. 2, pp. 285-297, May 2019.

[13] Y. M. G. Costa, L. S. Oliveira, A. L. Koericb and F. Gouyon, "Music genre recognition using spectrograms," *2011 18th International Conference on Systems, Signals and Image Processing, Sarajevo,* pp. 1-4, 2011.

[14] Y. Xu, Q. Kong, W. Wang and M. D. Plumbley, "Large-Scale Weakly Supervised Audio Classification Using Gated Convolutional Neural Network," *2018 IEEE International*

Conference on Acoustics, Speech and Signal Processing (ICASSP), Calgary, AB, pp. 121-125, 2018.

[15] Y. Su, K. Zhang, J. Wang and K. Madani, "Environment Sound Classification Using a Two-Stream CNN Based on Decision-Level Fusion," *Sensors, Basel,* vol. 19, no. 7, Apr 2019.

[16] M. Esmaeilpour, P. Cardinal and A. L. Koerich, "A Robust Approach for Securing Audio Classification Against Adversarial Attacks," *in IEEE Transactions on Information Forensics and Security,* vol. 15, pp. 2147-2159, Dec 2019.

[17] W. Deabes and A. E. Abdel-Hakim, "Teaming Up Pre-Trained Deep Neural Networks," *2018 International Conference on Signal Processing and Information Security (ICSPIS), DUBAI, United Arab Emirates,* pp. 1-4, 2018.

[18] X. Du, Y. Cai, S. Wang and L. Zhang, "Overview of deep learning," *2016 31st Youth Academic Annual Conference of Chinese Association of Automation (YAC), Wuhan,* pp. 159-164, 2016.

[19] S. Albawi, T. A. Mohammed and S. Al-Zawi, "Understanding of a convolutional neural network," *2017 International Conference on Engineering and Technology (ICET), Antalya,* pp. 1-6, 2017.

[20] Y. Chu, F. Huang, H. Wang, G. Li and X. Song, "Short-term recommendation with recurrent neural networks," *2017 IEEE International Conference on Mechatronics and Automation (ICMA), Takamatsu,* pp. 927-932, 2017.

[21] Y. Hua, J. Guo and H. Zhao, "Deep Belief Networks and deep learning," *Proceedings of 2015 International Conference on Intelligent Computing and Internet of Things, Harbin,* pp. 1-4, 2015.

[22] J. Wang, H. Lee, J. Wang and C. Lin, "Robust Environmental Sound Recognition for Home Automation," *in IEEE Transactions on Automation Science and Engineering,* vol. 5, no. 1, pp. 25-31, 2008.

[23] O. Gencoglu, T. Virtanen and H. Huttune, "Recognition of acoustic events using deep neural networks," *2014 22nd European Signal Processing Conference (EUSIPCO), Lisbon,* pp. 506-510, 2014.

[24] D. G. D. S. a. M. D. P. D. Barchiesi, "Acoustic Scene Classification: Classifying environments from the sounds they produce," *in IEEE Signal Processing Magazine,* vol. 32, no. 3, pp. 16-34, May 2015.

[25] S. Chachada and C. -. C. J. Kuo, "Environmental sound recognition: A survey," *2013 Asia-Pacific Signal and Information Processing Association Annual Summit and Conference, Kaohsiung,* pp. 1-9, 2013.

[26] T. Park and T. Lee, "Musical instrument sound classification with deep convolutional neural network using feature fusion approach," *arXiv preprint arXiv:1512.07370,* 2015.

[27] S. P. a. L. S. Soniya, "A review on advances in deep learning," *2015 IEEE Workshop on Computational Intelligence: Theories, Applications and Future Directions (WCI), Kanpur,* pp. 1-6, 2015.

[28] V. Boddapati, A. Petef, J. Rasmusson and L. Lundberg, "Classifying environmental sounds using image recognition networks," *Procedia Computer Science, Elsevier,* vol. 112, pp. 2048-2056, Sep 2017.

[29] C. B. R. Program, "Lab of Ornithology," Cornell Research, Canary 1.2.4 Software Program, 2001. [Online].

[30] R. S. Goldhor, "Recognition of environmental sounds," *1993 IEEE International Conference on Acoustics, Speech, and Signal Processing, Minneapolis, MN, USA, 1993,* vol. 1, pp. 149-152, 1993.

[31] J. I. Kazuo Hiyane, "Non-speech sound recognition with microphone array," *International Workshop on Hands-Free Speech Communication (HSC2001), Kyoto, Japan,* pp. 107-110, 2001.

[32] E. Dorken, E. Milios and S. Nawab, "Knowledge-Based Signal Processing Applications," *Prentice Hall Publications,* pp. 303-330, Jan 1992.

[33] M. Reyes-Gomez and D. Ellis, "Selection, Parameter Estimation, And Discriminative Training Of Hidden Markov Models For General Audio Modeling," *Proceedings of ICME-03, Baltimore, USA,* July 2003.

[34] L. Liu, "Ground Vehicle Acoustic Signal Processing Based on biological Hearing Models", Master's Thesis University of Maryland, College Park," *Master's Thesis University of Maryland, College Park,* 1999.

[35] T. Kohonen, Self-Organizing Maps., Berlin, Germany: Springer-Verlag . Printed in the U.S.A, 1997.

[36] S. Sampan, "Neural Fuzzy Techniques in Vehicle Acoustic Signal Classification," Master's Thesis, Virginia Tech, 17 Aug 1998. [Online]. Available: https://vtechworks.lib.vt.edu/handle/10919/30612. [Accessed 23 Feb 2020].

[37] J. Wang, C. Lin, B. Chen and M. Tsai, "Gabor-Based Nonuniform Scale-Frequency Map for Environmental Sound Classification in Home Automation," *in IEEE Transactions on Automation Science and Engineering,* vol. 11, no. 2, pp. 607-613, April 2014.

[38] L. Lu, H.-J. Zhang and S. Z. Li, "Content-based audio classification and segmentation by using support vector machines," *Multimedia Systems,* vol. 8, no. 6, p. 482–492, April 2003.

[39] P. Silva, ""“Classification, Segmentation and Chronological Prediction of Cinematic Sound," *2012 11th International Conference on Machine Learning and Applications, Boca Raton, FL,* pp. 369-374, 2012.

[40] M. M. Mostafa and N. Billor, "Recognition of Western style musical genres using machine learning techniques," *Expert Systems with Applications,* vol. 36, no. 8, p. 11378–11389, Oct 2009.

[41] L. Deng and D. Yu, "Deep Learning: Methods and Applications," *in Foundations and trends in Signal Processing,* vol. 7, no. 3-4, pp. 192-387, 2014.

[42] I. McLoughlin, H. Zhang, Z. Xie, Y. Song and W. Xiao, "Robust Sound Event Classification Using Deep Neural Networks," *IEEE/ACM Transactions on Audio, Speech, and Language Processing,* vol. 23, no. 3, p. 540–552, March 2015.

[43] H. Zhang, I. McLoughlin and Y. Song, "Robust sound event recognition using convolutional neural networks," *in Acoustics, Speech and Signal Processing (ICASSP), 2015 IEEE International Conference on, ,* p. 559–563, 2015.

[44] K. J. Piczak, "Environmental sound classification with convolutional neural networks," *in Machine Learning for Signal Processing (MLSP), 2015 IEEE 25th International Workshop on,* pp. 1-6, 2015.

[45] K. J. Piczak, "ESC: Dataset for environmental sound classification," *In Proceedings of the 23rd ACM international conference on Multimedia, ACM,* pp. 1015-1018, 2015.

[46] J. Salamon, C. Jacoby and J. P. Bello, "A dataset and taxonomy for urban sound research," *in Proceedings of the ACM International Conference on Multimedia, ACM,* p. 1041–1044, 2014.

[47] A. Krizhevsk, I. S. and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," *in Advances in neural information processing systems,* p. 1097–1105, 2012.

[48] C. Szegedy et al., "Going deeper with convolutions," *in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Boston, MA,* pp. 1-9, 2015.

[49] J. Salamon and J. P. Bello, "Deep Convolutional Neural Networks and Data Augmentation for Environmental Sound Classification," *IEEE SIGNAL PROCESSING LETTERS,* vol. 24, no. 3, pp. 279-283, March 2017.

[50] S. Hershey et al., "CNN Architectures for Large-Scale Audio Classification," *2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), New Orleans, LA,* pp. 131-135, 2017.

[51] J. F. Gemmeke, D. P. W. Ellis, D. Freedman, A. Jansen, W. Lawrence, R. C. Moore, M. Plakal and M. Ritte, "Audio Set: An ontology and human-labeled dartaset for audio events," *in IEEE ICASSP 2017, New Orleans,* 2017.

[52] Z. Kons and O. Toledo-Ronen, "Audio Event Classification Using Deep Neural Networks," *in Proceedings of INTERSPEECH,* pp. 1482-1486, 2013.

[53] A. Khamparia, D. Gupta, N. G. Nguyen, A. Khanna, B. Pandey and P. Tiwari, "Sound Classification Using Convolutional Neural Network and Tensor Deep Stacking Network," *in IEEE Access,* vol. 7, pp. 7717-7727, 2019.

[54] J. B. Wilson and J. Mosko, "A comparative analysis of whispered and normally phonated speech using an LPC-10 vocoder," *NASA STI/Recon Technical report Research Laboratory, Rome, NY,* vol. 86, p. 26504, Nov 1985.

[55] S. T. Jovicic, "Formant Feature Differences Between Whispered And Voiced Sustained Vowels," *Acta Acustica united with Acustica,* vol. 84, no. 4, pp. 739-743, July 1998.

[56] S. J. Wenndt, E. J. Cupples and R. M. Floyd, "“A study on the classification of whispered and normally phonated speech," *Seventh International Conference on Spoken Language Processing (ICSLP),* p. 649–652, 2002.

[57] C. Zhang and J. H. Hansen, "Analysis and classification of speech mode: whispered through shouted," *Eighth Annual Conference of the International Speech Communication Association (Interspeech),* p. 2289–2292, 2007.

[58] Z. Raeesy, K. Gillespie, C. Ma, T. Drugman, J. Gu, R. Maas, A. Rastrow and B. Hoffmeister, "LSTM-based Whisper Detection," *2018 IEEE Spoken Language Technology Workshop (SLT), Athens, Greece,* pp. 139-144, Sep 2018.

[59] I. Lezhenin, N. Bogach and E. Pyshkin, "Urban Sound Classification using Long Short-Term Memory Neural Network,," *2019 Federated Conference on Computer Science and Information Systems (FedCSIS), Leipzig, Germany,* pp. 57-60, 2019.

[60] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, A. C. Berg and L. Fei-Fei, "ImageNet Large Scale Visual Recognition Challenge," *International Journal of Computer Vision (IJCV),* vol. 115, no. 3, pp. 211-252, 2015.

[61] 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.

[62] K. Simonyan and A. Zisserman, "Very Deep Convolutional Networks for Large-Scale Image Recognition," *arXiv preprint arXiv:1409.1556,* 2014.

[63] X. Zhang, X. Zhou, M. Lin and J. Sun, "ShuffleNet: An Extremely Efficient Convolutional Neural Network for Mobile Devices," *arXiv preprint arXiv:1707.01083v2,* 2017.

[64] Iandola, F. N., S. Han, M. W. Moskewicz, K. Ashraf, W. J. Dally and K. Keutzer, "SqueezeNet: AlexNet-level accuracy with 50x fewer parameters and <0.5 MB model size," *arXiv:1602.07360,* 2016.

[65] M. A. Hearst, S. T. Dumais, E. Osuna, J. Platt and B. Scholkopf, "Support vector machines," *in IEEE Intelligent Systems and their Applications,* vol. 13, no. 4, pp. 18-28, July-Aug. 1998.

[66] J. Laaksonen and E. Oja, "Classification with learning k-nearest neighbors," *Proceedings of International Conference on Neural Networks (ICNN'96), Washington, DC, USA,* vol. 3, pp. 1480-1483, 1996.

[67] W. Contributors, "MATLAB," Wikipedia, The Free Encyclopedia, 8 February 2020. [Online]. Available: https://en.wikipedia.org/w/index.php?title=MATLAB&oldid=939727912. [Accessed 21 February 2020].