# Intelligent Sound Localization and Recognition with Tactile Feedback System to Assist Hearing-Impaired Individuals

by

Radia Tasnim Arony
21201209
Naomi Afrin Jalil
21201325
Maisha Raidah Chowdhury
21201560
Umma Hafsa Tahsin
21201523
Mohammad Shabab Bin Hassn
21201285

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

Department of Computer Science and Engineering
Brac University
February 2025

# Declaration

It is hereby declared that

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

**Student's Full Name & Signature:**

Radia Tasnim Arony
21201209

Naomi Afrin Jalil
21201325

Maisha Raidah Chowdhury
21201560

Umma Hafsa Tahsin
21201523

Mohammad Shabab Bin Hassn
21201285

# Approval

The thesis/project titled "Intelligent Sound Localization and Recognition with Tactile Feedback System to Assist Hearing-Impaired Individuals" submitted by

1. Radia Tasnim Arony(21201209)
2. Naomi Afrin Jalil(21201325)
3. Maisha Raidah Chowdhury(21201560)
4. Umma Hafsa Tahsin(21201523)
5. Mohammad Shabab Bin Hassn(21201285)

Of Spring, 2025 has been accepted as satisfactory in partial fulfillment of the requirement for the degree of B.Sc. in Computer Science on February 01, 2025.

**Examining Committee:**

Supervisor:
(Member)

Dr. Farig Yousuf Sadeque
Associate Professor
Department of Computer Science and Engineering
Brac University

Head of Department:
(Chair)

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

# Abstract

A significant portion of the global population suffers to some extent from hearing loss and requires assistive devices. These hearing-impaired individuals' everyday life is hampered due to not being able to recognize auditory cues in their surroundings. This thesis proposes a device that can both localize and identify types of sound while providing tactical feedback to the users, to make individuals with hearing impairment more aware of their surroundings. We have utilized machine learning models like Convolutional Neural Network CNN to identify specific sounds like car horn, dog barking, and scream. Furthermore, localization algorithms will be integrated into the device to identify the location of these sound sources and also notify the see-through readable text. Once these sounds are identified and localized, the user will be notified through vibration from a device. Both the direction and the type of the sound will be displayed to the user on the device. This proposed system will be a compact, low-cost, and practical solution to identifying auditory cues and will make a significant contribution to the field of assistive devices by improving the safety and life quality of hearing-impaired individuals.

**Keywords:** Assistive Devices; Sound Localization, Speech Recognition, Hearing Impairment; Wearable Technology; Tactile Feedback; Auditory Cues; Real-time Feedback

# Table of Contents

# List of Figures

# Chapter 1

# Introduction

Hearing impairment poses a significant problem, as a great number of people become victims to some extent of it. Due to sound pollution, this number will keep increasing. Hearing impairment especially causes difficulty in recognizing auditory cues from their surroundings. This limitation causes hearing-impaired individuals not to be fully aware of their environment which leads to a decrease in their quality of life and also poses a danger to their safety. Many assistive devices were created to address some of these issues. However, they are very costly and multiple adjustments have to be made which is not very user-friendly. Also, most of them only focus on amplifying the volume of these auditory cues without providing any contextual information, such as the location or content of these auditory cues.

Our proposed device will provide both the direction and the content of human speech. Speech recognition and sound localization technologies are integrated into the device which will allow hearing-impaired individuals to identify the direction of the voice and as well the read the content of the voice from the wearable device. The device will also provide tactile feedback through vibrations whenever a human voice is detected. The voice input will be taken by a microphone and advanced algorithms will be used to analyze the location of the voice and convert those spoken words into readable text. Once both of these processes are complete the user will feel a vibration on their wrist to be notified that a human voice has been identified. Users will be able to see the direction of the voice source on the display of the device. The content of the screen will also appear on the display and it will be in a scrolling motion. So, even if the content of the speech is long the user will be able to read it. This wearable device will be compact and low-cost making it a practical solution for identifying human voice. The purpose of the device is to improve the lives of people with hearing impairment by giving them the ability to be fully aware of their environment. This proposed system will make an immense contribution to the field of assistive technology by connecting sound recognition with meaningful feedback.

## 1.1 Research Statement

This research aims to find a better solution for hearing-impaired people who struggle in their day-to-day lives. With the perfect implementation of voice localization & speech recognition, this research wants to create a prototype of a hearing device that can be a reliable option for the people who need it most.

## 1.2 Research Objectives

1. Producing an alternative to 'Hearing Aid' which is the only solution for hearing impaired people.

2. Reducing the day-to-day hassle of a person with hearing impairment.

3. Combined application of Speech recognition & voice localization which is an important tool for smart voice assistants.

4. Method of building a tactile feedback system which is helpful technology for any smart device.

5. Designing a user-friendly and cost-effective assistive device that can be used by a wide range of hearing-impaired individuals.

# Chapter 2

# Related Work

## 2.1 Survey Methodology

We started collecting papers by focusing on keywords that are related to our topic. At first, we collected numerous papers with the help of Google Scholar & Semantic Scholar. Then we measured their eligibility by focusing on their proper relevance with our research, date of publication, where it was published & citation. After doing the first filtration, then we focused on the technology they used & tried to understand in depth so that we could figure out if that would be a feasible option for us to apply in our research. Thus we narrowed down our final list of papers. We worked with papers that offered voice localization with the help of wall reflection or electronic mapping or CNN. Multi-speaker speech recognition, keyword search from speech, electro-hepatic hearing, audio to tactile sensory, and recognition in spatially separated noise- these are the overall topics we covered. Finally, we summarized all the outcomes in chronological order which fits with our research work perfectly.

## 2.2 Related Work

Saxena, A., & Ng, A. Y. (2009) [1] experimented with finding sound location from only one microphone using an artificial pinna(outer ear). Sound localization using an array of microphones has been heavily studied already. However multiple microphones can make the system too big for practical use. A single microphone system is compact, power efficient, and lower cost compared to a multiple microphone system. The authors expressed that the experiment is inspired by biological monoaural systems. They used the Hidden Markov Model (HMM) and various publicly available datasets. Test data included human speech, dog barks, ocean waves, etc. The researchers have worked with four different artificial pinna designs from which two performed much better than the others. The best-performing pinna has smooth groves and the second best one has 2 sharp corners and some notches. The pinnas perform best for wideband noise(for example: radio static) and natural continuous noise (for example: ocean waves). This system can properly not localize pure tones just like actual human ears. The best pinna faces the least error when identifying widespread noise such as radio static. In this best-case scenario, the error is only 4.3° and the other pennas' errors are 8.8°, 22.3°, and 42.6°. For multiple overlapping speeches, the error in the best pinna was 7.7° and the second best gave 19.3°. For natural short sounds such as dog, bark performance is comparatively worse than

other sounds with an error of 14.2° for the overall second best penna and 18.3° for the overall best penna. The best artificial pinna had an average error of 13.5° which indicates that their compact model is fairly accurate at predicting the location of sound source.

An innovative method was proposed by Lim et al.[2]for speaker localization in noisy environments called Steered Response Voice Power (SRVP). Using this method, the location of human speech can be detected even in noisy environments for speech-oriented devices like TVs and humanoid robots. The problem with conventional algorithms is that they often make errors in localizing human speech because conventional algorithms like SRP-PHAT consistently recognize the source of the loudest sound. This is problematic because, in a noisy environment, the loudest sound might not come from a human. However, the proposed SRVP method solves this issue by combining SRP-PHAT with Voice Activity Detection (VAD), resulting in being able to distinguish between human speech and other noises. As a result, Steered Response Voice Power (SRVP) can detect the location with the loudest voice power. The new proposed algorithm first uses SRP-PHAT to find all potential sound sources to pick top candidates from it. Then with the help of VAD, the source with the most "voice-like" characteristics is found. As a result, human speech can be localized even in noisy environments. There were two types of tests that were conducted. One with simulated data and another with real-world recordings using a robot with a microphone array. A combination of simulated room acoustics data and real-world recordings were used in these tests. In both of these tests, the researchers used a circular array of microphones containing eight microphones and placed them in a simulated room for the simulated test and a real room for the real data test. The microphones were surrounded by one speaker which was the voice source and multiple noise sources were placed at various points in the room. The speaker localization algorithm would be successful if it could identify the speaker's location within ±5 degrees of its real position. The test also used various Signal-to-Noise Ratios (SNRs) ranging from -5 dB to 15 dB. The higher the SNR value the louder the human voice will be than the noise. The SRVP algorithm performed better than the traditional SRP-PHAT at every noise level. For example, at 0 dB SNR which meant both human voice and noise level were equal, only 18.8 % accuracy was achieved by a conventional algorithm SRP-PHAT. Whereas, the new algorithm was able to achieve an accuracy of 30.6%.The SRVP method also achieved superior results in tests where real data was used. In this test at 0 dB, SNR SRP-PHAT had an accuracy of 23.7% whereas SRVP achieved 42.9% accuracy. Thus, the new method SRVP is a much more robust and suitable speaker localization method than conventional algorithms like SRP-PHAT.

Archana et al. [3] developed a compact device to aid visually impaired patients by detecting nearby obstacles and alerting them. The device uses special sensors to detect these obstacles like sonars used by bats and once these obstacles are detected, it sends notifications to the users through vibration, beeps, and voice indications. As a result, an improvement in the mobility of hearing-impaired individuals can be seen due to the device's ability to make them more informed about their environment. An ultrasonic sensor (HC-SR04 model) is used to detect obstacles and the obstacles are detected by calculating the time taken for the high-frequency sound

waves sent out by the sensors to bounce back from the obstacle. This also helps to calculate the distance to the obstacle. The signals from the sensor are processed by an AT89C51 microcontroller. The device also provides voice messages which are used to notify users when obstacles are detected. Additionally, two types of buzzers are also present to alert the users when obstacles are detected. All the components were assembled by the researchers and placed in a setup designed to mimic real-world conditions for testing. The sensors of the device receive a trigger signal and if an obstacle is detected by the sensors, the sound buzzer or voice alert is activated by the microcontroller to notify the user. Two types of buzzers continuous and intermittent were used to indicate whether an obstacle was on the left or right side. For example, the buzzer with the continuous sound would activate if the obstacle was on the right side and the buzzer with the intermittent sound would activate if the obstacle was on the left. While alerts for obstacles directly in front are provided by voice commands. Obstacles can be detected as long as its within a range of 0.3 to 4 meters and a 60-degree field of view. Expected results were seen when tests were conducted in a controlled environment. The device can both accurately detect and notify users of nearby objects according to the object's direction.

Audhkhasi et al. (2017) [4]built a model for end-to-end(E2E) keyword search from speech without using automatic speech recognition(ASR). This model is much faster than ASR-based keyword search (KWS) models, it does not need training data from the target language to search words from that language and it can be trained with multilingual data for increased performance for many languages. Their model detects the occurrence of out-of-vocabulary (OOV) words as well as in-vocabulary (IV) words from the input. The model has three major components. 1)Acoustic input is encoded to a vector using recurrent neural network(RNN) 2) RNN and convoluted neural network(CNN) is used to embed a regular input text query so it can be used to find its occurrence in speech 3)The encoded sound input and the embedded text query is compared to find if the keyword occurred. The RNN does not need any transcribed speech data for training, it also does not need any data from the target language. They tested many strategies for the acoustic autoencoder to minimize mean-squared error (MSE). They tested the sound encoder training using different strategies to find the smallest MSE. From the 4 strategies, they found that batch size = 80 reduces MSE the most. Frame stacking and decimation also significantly decreased training time and computation load for RNN. The ASR free KWS finds the occurrence of OOV words better than KWS systems that use ASR. The ASR free model has an accuracy of 70.77% for OOV words whereas the ASR-based model has an accuracy of 50%. The ASR-based model always gives 0 (did not occur) output for OOV words so, the 50% is basically blind chance. For IV words ASR free model still performed a little worse than the ASR using model however the researchers mentioned this was an improvement from their previous work. Finally, the proposed model tried to find the time when the keyword occurred which shows the potential of time localized keyword search.

Vecchiotti et al. [5]proposed a complete data-driven approach for detecting voice activity & localizing them in the enclosed environment. Such speech detection & speaker localization were performed with the help of a combination of Neural Network (NN) based Voice Activity Detection (VAD) and an NN based Speaker LO-

Calization (SLOC). It was a single data-centric system which was called the "Joint VAD-SLOC Model". The cascade structure was built by training a Neural SLOC which actually trained an Oracle VAD whose task is to select only the speech portion of the audio signal. VAD helped to evaluate SLOC performance. A convolutional Neural Network (CNN) was used which takes two different features as inputs: Log-Mel & GCC-PHAT. A single CNN processed these two features with the help of two stacks of convolutional layers. Then some fully connected feed-forward layers gave three outputs: prediction of voice activity & two coordinates which determined speaker position. In total, the Joint VAD-SLOC Model produced three outputs and they were the combination of Neural VAD & Neural SLOC output. As this method took both speech & non-speech signals, binary labeling was used for Neural VAD to identify between them. For Neural SLOC, the 2 outputs that upheld the coordinates were in the range of 0 to 1. The Activation function was the ReLU activation and another specific function was employed for this specific model which is called Hard Tanh which represents f(x)=x where x is in the range from 0 to 1. All algorithms are written in Python using Keras, training used standard backpropagation, the optimizer was Adam, and early stopping & variable learning rates were imposed. The DIRHA dataset was used which was taken with the help of 40 microphones installed in a five-room apartment. Between two consecutive microphones, there was a 50 cm distance maintained. LogMel Features were evaluated from all the available microphones (for neural VAD) whereas GCC-PHAT features were evaluated from all the couples of adjacent microphones (for neural SLOC). A 10-fold cross-validation was employed. Among them, 8 were for training, 1 for validation, 1 for testing. They evaluated evaluation metrics which held false alarm rate(FA), deletion rate (Del) & speech activation detection (SAD) for VAD evaluation. For localization accuracy, RMSE & P(cor) was calculated. When calculating the accuracy of speech detection, the proposed model showed a reduction in the average value of SAD error which is a 33% straight reduction. The combined configuration showed a 2.7% relative increase in the value of RMSE compared to the Neural SLOC.

Mahamud and Zishan [6] developed a wearable device to assist hearing-impaired people by converting human speech to readable text in real-time communication. The device operates by taking voice input through an app on the phone which is then converted to text. The converted texts are then sent to the wearable device via Bluetooth where it is displayed on an LCD screen on the wearable device for the user to read. The device has no limitation on speech input and uses Google's speech recognition technology to convert human speech to text. So, the device can understand any spoken English words, giving the user the opportunity to communicate freely with others. The system consists of a mobile app, a Bluetooth module, an Arduino Uno, and an LCD display. All these components work together to assist hearing-impaired people in communicating. The custom app named "ANKON" takes voice input and converts it to text. Then the app pairs with the wrist device and transmits the text data wirelessly. The Bluetooth module, called HC-05, is integrated into the device to enable this wireless communication. It passes the text data to the Arduino Uno which is the central processing unit of the wrist device. This microcontroller can efficiently handle the text data and requires a 5-volt power supply to operate. Once the text data is decoded by the Arduino Uno, it sends it to the connected LCD display where it can finally be displayed to the user. The

display size is small but clear, showing 16 characters per line over two lines. The display can show longer messages by scrolling the text on its screen. For instance, long messages like "Good job the project is successful" will scroll across the screen, allowing the users to read messages easily even during long conversations. The device is compact, user-friendly, lightweight, cost-effective, and secure against hacking. During testing, they perform well by accurately converting human speech to text. In both simulations and real-world conditions the device was able to convert voice inputs into text and display them on the screen. There are future plans to support multiple languages, allowing users to communicate in their native language. The device has the capability to become a practical solution to improve the life quality of hearing-impaired individuals.

A. H. Michaely et al. [7] suggested a system that will improve the overall accuracy of the Keyword Spotting (KWS) system by reducing the False Accept rates and also minimizing the False Reject rates. KWS is a widely used system for hands-free voice activation by detecting specific trigger phrases. To develop such system the researchers have proposed two-tiered KWS system where the first or primary KWS will operate locally with limited computational resources to detect trigger phrases. When a phrase that can be a potential trigger is detected, it will send the entire audio including the trigger phrase and subsequent query to the server. Then the secondary KWS or the server-side KWS system will utilize larger ASR models and more advanced algorithms to reanalyze the audio and decode both the trigger phrase and query audio using a context-aware recognizer. If the decoded audio does not contain the trigger phrase, the query is suppressed which means the false accepts will be eliminated. By doing this it will improve certain speech characteristics such as coarticulation effects of speech and noise handling. They also suggested additional features which is trigger phrase detection from the transcription once detected and remove it to ensure smooth processing. It also ensures a second pass on the severside system where the results of the first decoding are reanalyzed to further reduce error rates. After testing their method in the field the system achieved a significant reduction in false accepts (FA) by 89% but it also increase false rejects(FR) by 0.2% which is very minimal in comparison to the increase in false accepts so the overall performance still improves significantly. This method also improves Automatic Speech Recognition quality by reducing the Word Error Rate by 10-50%. It is also effective for different languages which shows its broader applicability.

Yağanoğlu and Köse [8] developed a wearable device for real-time detection of important sounds for hearing-impaired individuals. Many challenges in their daily life are faced by these individuals due to their inability to recognize crucial sounds, which can sometimes result in putting their lives at risk. The device aims to enhance hearing-impaired individuals' quality of life by helping them recognize these important sounds. Various sounds like doorbells, alarms, phone rings, car horns, and more can be identified by this device with the help of audio fingerprinting and notifying the users through vibration. There are four main components in the system which are Raspberry Pi, grove Shield, microphone, and vibration motor. The microphone is used to pick up sounds, which are then received by a Raspberry Pi to be processed. Then with the help of audio fingerprinting, Raspberry Pi identifies the type of sound. Once identified, the device sends a specific vibration pattern through

a motor to notify the user about the identified sound. A dataset of 10000 sound samples which had eight categories such as doorbells, alarms, car horns, phone rings and etc was used to train and test the system to identify the type of sound accurately. Different techniques are used to recognize and differentiate between these sounds like Zero Crossing Rate (ZCR), Mel Scale Cepstrum Coefficients (MFCC), Line Spectral Fre, K Nearest Neighbors (KNN), Log Area Ratio (LAR), Linear Prediction Coefficients (LPC), and Audio Fingerprint. Here, the Audio Fingerprint technique creates a unique "fingerprint" of each sound by observing its unique characteristics like the peak in frequency. So, when a new sound is detected it compares against the stored fingerprint to identify the type of sound. The system uses all the methods to quickly and accurately identify sounds and notify the users even in noisy environments. The device was tested both indoors and outdoors and it used different levels of vibrations for each type of sound. It was able to identify simple sounds like phone rings and alarms around 99% of the time and complex sounds like cars honking around 91-93% of the time. Additionally, the users were notified within 1.9 seconds. Though noisy environments caused the performance decrease, it still outperformed other sound recognition methods. Users have also given feedback that this device could improve the lives of hearing-impaired patients by making them more attuned to their environment.

Seki et al. (2018) [9] proposed an end-to-end sequence-to-sequence framework to separate and recognize multiple speakers from one input channel without having distinct source separation and speech recognition stages. Their model does not need isolated source data for reference and it has an objective function to increase the contrast of hidden vectors for different speakers which improves its performance. The authors explained that previously speech recognition for multi-speakers was done using two-step processes where at first separation was executed and then recognition was executed. Previous models that do not require explicit modules for separation and recognition still need isolated sources to use as reference which is not required for their proposed model. The model uses an attention-based encoder-decoder to predict a label from speech. These labels become labels after decoding, for the encoder it is called hidden representation. To find an output label context vector, and CTC loss are used. A context vector is calculated using the attention mechanism, which gives more weight/attention to parts that are important for label prediction. Permutation invariant training(PIT) is utilized to get multiple predicted labels for multiple speakers from a single-channel mixed speech input. PIT ensures correct label sequence is assigned to each speaker. The Decoder module outputs a label sequence for each speaker by independently decoding each sequence's hidden vectors. An objective function is applied that uses Kullback-Leibler(KL) divergence to increase the contrast of hidden vectors. To split hidden vectors VGG and bi-directional long short-term memory (BLSTM) are used and compared. The experiment was done in the English language using the WSJ dataset and in the Japanese language using the CSJ dataset. The experiment dataset is made by mixing the speech of two speakers(each taken from the existing WSJ/CSJ corpus) with varying levels of sound-to-noise ratios. First, they used the model for unmixed data which gave a character error rate(CER) for English 2.6% and for Japanese 7.8%. Here training was done for unmixed/single speakers and the test was done on the same simple case so, this was considered the lower bound for CER. For mixed speech in English

baseline/no hidden vector split gave an average of 83% CER, the VGG split gave 16.5%, the BLSTM split gave 14% CER, and BLSTM with KL loss gave the lowest error which is 13.7%. So, clearly, the BLSTM split with high hidden vector contrast performed the best. For mixed Japanese speeches, worst case/baseline CER was 92.7%, and best case BLSTM with KL loss gave 14.9% CER. Compared to other models that have explicit separation (using DPLC) and recognition (using ASR) the proposed model by Seki et al. **<empty citation>**gives a 2.6% reduction in word error rate (WER).

Fletcher et al.[10] conducted a study on electrohaptic stimulation on Cochlear Implant(CI) users, to prove that haptic feedback on the skin improves speech intelligibility in noisy environments that CI users struggle the most in. The participants underwent speech testing before and after training sessions. To test and train them two speech corpora that were unique from each other were used respectively (i)The BKB Institute of Hearing Research corpus and (ii) RealSpeech™(UK) content library. RealSpeech material contained a variety of general-interest topic narratives along with multi-talker noise that was non-stationary recorded by National Acoustics Laboratories(NAL); these speech and noise signals were converted to tactile signals using signal processing. The signal processing makes use of tactile signal frequency ranges from 50-230 Hz which is easily produced by low-power devices, then downsampled to 16 Hz as sampling frequency and passed through a 4-channel FIR filter bank divided equally as on ERB scale, after which Hilbert envelope was computed for each channel and low pass filter of first order was applied. Four fixed tonal carriers' amplitude envelopes were modulated for the channels, the carrier signals were then individually passed through an expander function to reduce noise and amplify temporal modulation therefore increasing the prominence of speech. The 10 participants were tested and trained with four sessions, 2 testing, and 2 training, to do this they were placed in a sound-attenuated booth, where stimuli were created and controlled by MATLAB scripts from a laptop in a different room. Acoustic sounds were produced by RME Babyface Pro soundcard, Genelec 8020 PM Bi-amplified Monitor System, positions of the stimuli were calibrated using Brüel & Kjær (B&K) G4 type 2250 sound level meter and B&K type 4231 sound calibrator. For vibrotactile stimuli, two HVLab tactile-modified vibrometers were used. The overall results were in favor of the experiment even though they had a lesser improvement of 8.3% compared to other studies were 10.8% but these had different conditions such as placement of vibrometers, duration of testing, age group, the mean improvement of the study was 8.3% points with the largest improvement of 21.8%points. It is to be noted the best performer had previous experience with tactile training due to playing the flute and the smallest improvement was 0.5% points.

Chang et al. (2020) [11] conducted a study using neural sequence-to-sequence (seq2seq) architecture to recognize and separate speech in a multichannel, multispeaker input system and give multiple output text sequences for different speakers. The experiment was done on a reworked version of the WSJ1 dataset which is specialized for multispeaker speech training as it contains noise and overlapping speech of different speakers. The proposed model was built using Pytorch and it is based on the ESPnet framework. It has three major layers for separating and recognizing speech from the microphone array input. Firstly, for each speaker of

each channel(microphone) of the multichannel input, a time-frequency mask is estimated. Short-term Fourier Transformation(STFT) is used to define a set of masks for each input channel. Each set is made of masks for each speaker, and an additional mask is created for the background noise as well. Secondly, the Power Spectra Density (PSD) matrics of each speaker and noise are calculated from the masks. Using this beamformer's filter co-efficient is calculated at a frequency for each speaker. Other speakers' PSDs are considered as noise when filter coefficient of a speaker is calculated. The neural beamforming filters are applied to the multichannel multispeaker inputs to get separate noise-free outputs for each speaker. Lastly, the outputs from the neural beamformer are normalized and fed into an end-to-end speech recognition submodule. The submodule uses permutation invariant training(PIT) for input-reference sequence assignment. Curriculum learning was deployed for improved training of the model. Their model is not only trained on multi-channel,multi-speaker datasets but also single-speaker, single-channel datasets. The report contains results for instances of two speakers and two input mics, but the model can accommodate any number of input channels. Traditional beamforming algorithms do not perform well for overlapped speech signals because they can not properly separate the speech of different speakers. The proposed model(named MIMO speech mode)performs 60% better than the baseline models. MIMO speech model with curriculum learning gave a character error rate(CER) of only 3.75% and a word error rate (WER) of only 7.55% for multi-speaker speech recognition(anechoic). The model performed relatively worse(12%) when they did not use curriculum learning for training the model. However, curriculum learning does not improve performance of separating the speech of different speakers. The perceptual evaluation of speech quality(PESQ) and signal-to-distortion ratio(SI-SDR) indicate that the neural beamforming successfully differentiates two speakers. The MIMO speech model performs well in terms of speech recognition and separation for multi-channel multi-speaker anechoic inputs.

Shen et al. [12]proposed an algorithm-based approach for localizing any human or object in an environment. They tried to calculate the Angle of arrival (AoA) with the help of one microphone array. The main challenges were to estimate the AoA without the knowledge of the source signal and trace back these AoAs for user location. They focused on two things primarily: accurate AoA detection & joint wall geometry estimation. To accurately detect the AoA, they developed a new algorithm named Iterative Align-and-cancel (IAC) which performs both alignment & cancellation to find AoAs. Here it is different from the existing algorithms which only perform the alignment. This algorithm cancels the correct AoAs that are being repeated and the belated AoAs that have traveled longer than the direct path (mainly echoes or reflections). The system architecture allows two AoAs (AoA1 & AoA2) from the raw sound. Then it takes 3 parameters: wall distance & wall orientation from the wall geometry estimation. Also the user's height. Along with these 3 geometric parameters and AoAs, the fusion model produces the location. To implement it in real life, they used a Seeed studio 6-microphone array whose acoustic signals are at 16 kHz. The array is connected to Raspberry Pi which is connected to the laptop to execute the codes. To test the model, 4 different indoor environments were used: a studio apartment, kitchen, student office, and large conference room. For the limitation of the model, they used an environment with no obstacle, the source

is situated at a static height, standing & not moving. Also, the source is within 4-5 m from the device. A total of 2350 voice commands &some non-voice recorded sounds were used to acquire the result. When comparing the performance, the IAC algorithm has significantly better performance than common algorithms like GCC-PHAT, MUSIC, and Delay-and-Sum in correctly detecting AoA. It can be a new contribution to the field of AoA algorithms. The overall median error of the whole model to localize a user is reduced by 52% than GCC-PHAT's median error. The overall median error is 0.44 m. Among different environments, the conference room has higher errors because of its larger size. Among variations, localizing objects is easier than humans. A closer location produces higher accuracy. The model is not sensitive to the user's voice or speaking pattern but it is sensitive to clutter levels & background noise. Dense environments offer higher median error rather than sparse ones. Likely, constant background noise increases errors in sound source localization.

Haoheng Song et al. [13] suggested a novel solution of using "electro-haptic" stimulation, which combines electrical auditory stimulation provided by the cochlear implant (CI) with haptic feedback delivered to the wrists for hearing-impaired individuals using cochlear implants (CI). While cochlear implants restore the hearing of a profoundly hearing-impaired person largely, there remains poor performance by users in environments with multiple talkers or background noise; difficulties in understanding speech in noisy environments remain a challenge. Poor performance by users remains in environments with multiple talkers or background noise.. These issues affect their ability to perform effectively in real-world listening environments, where distinguishing speech from noise is crucial. The solution will be a means of restoring missing speech and spatial hearing cues through haptic signals, which will supplement the auditory information provided by the CI. This haptic stimulus will also provide additional speech and spatial hearing cues beyond what would have been conveyed by the CI in isolation. To this end, they suggested a non-invasive haptic system that could be implemented as a wearable neuroprosthetic device- low-cost and easy to integrate with current CI technology. It would greatly enhance the ability of CI users to perceive speech in noisy conditions, offering a radical improvement over the existing solutions. This system yielded large improvements in understanding speech in a noisy place with many speakers after just 30 minutes of training, especially when the speech and noise came from different spots. They achieved a 2.8 dB average improvement and 2.6 dB for SRT for noise coming from the same side and noise coming from the opposite side in this study. This gain was observed across three unique cochlear implant systems of users, showing the various ways EHS can be applied. Being able to function with different systems suggests that this solution can be adjusted for most types of CI technology. Consequently, this report presents an innovative and efficient technique for improving cochlear implant performance with the incorporation of haptic feedback to enhance speech recognition in adverse listening conditions like the real world for CI users and could improve the lives of hearing-impaired patients by making them more attuned to their environment.

Kim et al. [14] in their paper thoroughly discussed several algorithms of end-to-end (E2E) speech recognition systems, components, and optimization methods for on-device applications. Conventional speech recognition systems require too many individual pieces such as language model, acoustic model, pronunciation model,

Weighted Finite State Transducer(WFST) decoder and so on and so forth. End-to-end neural speech recognition performance surpasses WFST-based decoder or n-gram Language models with large training datasets and has better Byte Pair Encoded(BPE) subpar units. Gated RNNs have smaller memory footprint requirements compared to other E2E models along with their streaming capabilities if unidirectional. Several CNN-based models use nonlinear rectifiers and groups of filters but unlike Gated RNN, CNN does not need to compute previous time steps for the current one and such have efficiency when using intrasequence parallelism if Single Instruction Multiple Data (SIMD) is supported. ERE Architecture included (i) Connectionist Temporal Classification(CTC) -uses layers of neural networks in a stack with CTC loss, parameters to update CTC loss are similar to Hidden Markov Models(HMMs) and use forward-backward substitution assuming conditional independence. (ii) Recurrent Neural Network Transduce(RNN-T) -uses a feedback path that is explicit from the output label to the prediction network which improves on CTC's lack of explicit feedback path and assumption of conditional independence, (iii) Attention-based Model- uses attention between decoder and encoder hidden outputs, (iv) Monotonic Chunkwise Attention(MoChA)-based model- uses two mechanisms the hard monotonic attention than soft chunks attention. Among compression techniques simplest is using TensorflowLite™ 8-bit quantization, others include Low-Rank Approximation(LRA) but the overall best approach is pruning if hardware capable, else distillation for small models, and factorization for large models. To improve performance (i) Nonstreaming mode combination, (ii) shallow fusion with Language Models(LM) and, (iii) NER performance with spell corrector were described.

Korayem et al. [15]talked about a method of voice command recognition with the help of Hidden Markov models (HMM). They mainly focused on how to come up with a low-cost system that will process single-word commands as similar workings have been achieved with more complicated systems. After detecting the voice command, they implemented it on a robotic face that could give facial expressions & on a scout robot that could move its robotic body according to the commands. As the movement of the body was directed to the source, they implemented a part of voice localization in this case. To achieve the mentioned idea, they used a dataset which is a nonnative database produced by Persian speakers containing 10 English commands. It was a small database that included 'ready',' forward',' backward', 'left', 'right', 'stop', 'move', 'faster', 'slower' & 'go'. These 10 control words were spoken by 18 speakers & they spoke each of them 3 times. So, the total database was built with 540 wave files where each speaker held 30 files. The format of the data was .wav & the sampling frequency rate was 16,000 Hz. The whole dataset was labeled manually. They took the help of GUI for recording the dataset & training the algorithm. They divided the dataset into training & testing in such a way that the speakers of the training set and the speakers of the test set were different (percentage not mentioned). The Hidden Markov model was used for the speech recognition part & for the voice localization, they used HARK, also known as a voice localization package. ROS middleware was used for the voice synthesis module to improve the robot interaction. The speech recognition system consisted of feature extraction, acoustic modeling & decoding phase. MFCC features were extracted in the feature extraction part. For acoustic modeling, each voice command

was separated into frames of 25 ms. As the phonetic unit is Word in this scenario, one HMM is trained for each word. 11 word-based HMM were trained for silence & 10 control words. They used a dictionary to identify the arrangement of the words. For the decoding phase, the Viterbi Algorithm was used. The duration to detect a single trained command was 0.9 s & the word error rate was 0%. For non-English speakers, the accuracy decreased from 100% to 85.16%.

Fletcher et al.[16] conducted an experiment in this paper to see if haptic stimulation can be used in 3 locations-(i) lower triceps, (ii) palmar wrist, and (iii) dorsal wrist as the tactile intensity differences(TIDs) could supplement electrical CI stimulation to help hearing-impaired listeners. The paper goes on to show that previous research mainly focused on fingertips and palmar region of people, the wrist is a practical position but may be occupied by other devices or TIDs can be distorted by movements, as such triceps were studied for discreteness and easy self-fitting. 12 participants were first screened with HVLab Vibrotactile Perception Meter, 6mm contactor with rigid sound; these were adjusted using a B&K calibration exciter. During testing custom MATLAB scripts were used to control custom Max 8 Patch for threshold measurements. Measurements in the testing phase were carried out by RME Babyface Pro soundcard, two HVLab tactile vibrometers were calibrated using the HVLab tactile vibrometer's accelerometers that are inside it. During the testing phase stimulus used was amplitude modulated and unmodulated, modulated used the dataset from the University of Southampton Research Data Management Repository using similar technology as Fletcher et al.[3], a three alternative forced choice paradigm was used, among the three intervals in each trial only one accompanied a signal that participants tried to identify, using a two-down one-up approach for threshold limb thresholds then repeated with a two-alternative forced paradigm for TID discrimination task. Repeated-measures analysis of variance(RM-ANOVA) was used for the TID discrimination and detection thresholds. The performance from the research showed the dynamic range averaged for dorsal, palmar, and lower tricep to be 55.6 dB, 57.6 dB, 54.3 dB, and TID detection threshold for left-right orientation respectively for dorsal wrist 0.0114 ms−ˆ2 and 0.0091 ms−ˆ2, for palmar wrist 0.0104 ms−ˆ2 and 0.0080 ms−ˆ2, and lower triceps 0.0116 ms−ˆ2 and 0.0137ms-ˆ2. Overall TID discrimination threshold was 1 dB for all locations, similar to across ear stimulation of normal hearing listeners, palmar wrist had the highest sensitivity but the average difference from other locations was (0.3 dB).

Hanan Aldarmaki et al. [17] proposed a system that can transcribe speech into text without relying on large labeled datasets or human-generated annotations in other words unsupervised ASR systems can achieve reasonable performance without labeled datasets. This will support languages with limited resources or even no written form with the understanding of what can be learned from raw speech signals in the absence of extensive resources. So they tackled the challenge of low resource Automatic Speech Recognition (ASR) and human language acquisition insights by segmentation of speech, feature extraction and embedding, Acoustic Unit Discovery, Cross-Modal Grounding, Unsupervised Sub-Word and Word Modeling, Iterative Refinement, and Pseudo-Supervised Learning. They try to replace large labeled datasets with unsupervised learning methods in order to construct a speech recognition system in low-resource languages. Most of the steps mentioned above

have been combined or approached simultaneously for better overall performance in the modern model. The phonetic segmentation model has given very excellent performance with the boundary F score above 80%. This is considered to be a milestone in unsupervised speech processing. The second problem, word segmentation, is significantly more difficult, and several promising directions of this review include the inclusion of collocations, pauses, and syllabic segmentation, which add much value to identifying word boundaries. Other exciting findings involve studies into a variety of methods for doing cross-modal mapping, including Generative Adversarial Networks, in order to align segments of speech and text in the absence of explicit supervision. Indeed, results under this approach have also been quite promising, especially those at the phonetic level. A couple of such models achieve word error rates as low as 11%, which would be huge steps toward closing this gap between unsupervised and supervised models. By analyzing the existing models and techniques, the researchers provide a clear roadmap for future research in unsupervised ASR. This also involves how a combination of bottom-up phonetic models with top-down semantic models will be improved, word segmentation improved, and further refinements made in speaker-invariance in embeddings.

Chen et al. [18]came up with the idea of creating Voicemap which provided a mapping system conducted autonomously to treat the purpose of Voice Localization. A sweeping robot served as their medium which is a part of smart home appliances. There was a cooperation between the robot & voice devices. The robot automatically explored the map of the environment & determined different positional information of the voice devices existing on the map. Thus they could localize the voice sources on the map. The microphone array did the localization of the robot, so there was a relation based on positional value between them. The main challenges were: accurately locating the constantly moving robot, finding pure samples of voice signals as the sweeping robot constantly made noises, and synchronizing the sweeping robot's coordinate system & the coordinate system of the microphone array. To solve these problems, to calculate the AoA (Angle of arrival), they implemented an inertial-based super-resolution method to localize the sweeping robot. It combined object motion & AoA estimation to increase accuracy with the help of time-frequency analysis. It developed a filter to get the intersection of numerous peaks. They also took the robot trajectory into consideration. After solving the AoA estimation problem, they synchronized the SLAM map created by the robot & AoA so that they could synchronize the coordinates of the moving object and the microphone array (device). After determining the location of the device in the SLAM map, they could finally perform voice localization. The prototype was built with COTS SLAM Robot, Seeedstudio Respeaker 4 mic linear array, 6 mic circular array. The array was controlled by the Raspberry Pi 4 model B. The sound sampling rate was 48 kHz. The performance of AoA estimation was higher than the normal rate which is 0.12 m which is 43.9% reduced than the art. The performance of mapping was 0.12 & 0.13 in two different scenarios. Finally, the overall performance of localizing humans has higher accuracy than both the Symphony model & MAVL model as they have higher median error than the proposed model. The proposed model has a median error of 0.22 m.

Fletcher et al. [19] explored in this paper by testing participants with tactile mo-

tion on the wrist if multiple frequency channels can improve tactile stimulation to discriminate phonemes (units of sound in speech) by transferring spectral information(representation of sound frequency components). They used audio-to-tactile vocoder signal-processing similar to Fletcher et al[3] except for higher frequencies 50-7000 Hz to get tactile frequencies that tactile systems are highly sensitive to, to be provided on a single site, this provided the largest effect to be seen for place contrasts and voicing. Performance is increased by multiple frequency channels for consonant pairs, especially the ones varied by voice(sound made by vocal cord vibrating) alone or voicing and place. 26 participants were tested in an isolated room in front of an EHS Research Group haptic simulation rig which was custom made of Ling Dynamic Systems V101 shaker, 3D printed circular probe with 10mm diameter and no rigid surround. The shaker was operated using a MOTU UltralLite-mk5 sound card, RME QuadMic II preamplifier, and HV Lab Tactile Vibrometer power amplifier. The probe was hung from a frame onto the dorsal wrist area where people usually wear watches and the vibration was adjusted using a B&K 4533-B-001 accelerometer and a B&K type 4294 calibration exciter. The testing was done using three intervals, three alternatively choice phoneme discrimination tasks, each trial made use of pair phonemes spoken by a single talker who spoke from the dataset of 53 paired phonemes that was converted to tactile stimuli, of which one phoneme was presented once the other twice and this was randomized, with stimulus intervals of 250ms and participants had to choose which was presented once. The percentage of phonemes was measured in 5 conditions-(i) one frequency band and one vibrotactile tone (1FB1T), (ii) one frequency band and four vibrotactile tones (1FB4T), (iii) four frequency bands and four vibrotactile tones (4FB4T), (iv) one frequency band and eight vibrotactile tones (1FB8T), and (v) eight frequency bands and eight vibrotactile tones (8FB8T). Results from the experiment showed improvement when using 4 bands compared to 1 band was on average 5.9% and for 8 bands was on average 8.4%. Larger improvement was seen for 8 bands than 4 bands especially in consonants against vowels, using 4 bands mean improvements for consonants and vowels were respectively- 8.6% and 3.1%, and mean improvements using 8 bands for consonants and vowels were respectively- 14.8% and 1.7%. Overall participants' scores differed by talker male- 44%, female- 48.7%; it was found between talker and phenome type(consonant and vowels), for male speakers performance in consonant-43.4%and vowels- 44.6%. For female speakers performance in consonants- 51.7%and vowels- 45.7%.

Mark D. Fletcher et al.[20] introduced an advanced dual-path recurrent neural network(DPRNN) as a core technology for noise reduction and speech enhancement using hearing-assistive devices and, more specifically, for CI users. They used data by conducting a behavioral experiment on 16 adults between 18 and 37 years. Here noise and speech masking were done with both non-stationary noise (party noise with ILTASS spectrum) and stationary noise filtered to match the ILTASS. A dual-path recurrent neural network method was used for background noise removal. In this case, the DPRNN segmented speech into chunks processed these through both bidirectional and unidirectional LSTM layers, and used masks for noise reduction in extracting clean speech. For online processing, the implemented quasi-causal DPRNN had a frame chunking system with a chunk size of 20, meaning that it could only see 5.75 ms into the future and, thus, can be made feasible for hear-

ing aid application purposes. The speech amplitude envelopes were extracted by Hilbert transform and filtered with a zero-phase 6th order Butterworth filter with a 23 Hz cut-off; this has removed the silences, thus making SNR more constant across sentences. The participants received vibrotactile stimulation through a Ling Dynamic Systems V101 shaker applied to the dorsal wrist with applied tactile tones which were equally spaced in frequency and adjusted for detectability based on thresholds. Quasi-causal DPRNN, non-causal DPRNN, and Causal DPRNN methods were trained on large datasets, ensuring their robustness across various noise environments and SNRs. While log-MMSE is widely used in hearing-assistive devices and performs well in stationary noise, it is less effective in non-stationary, real-world noises like the multi-talker scenario used in the study. Therefore, the DPRNN, especially the non-causal and quasi-causal versions, provided better noise reduction performance, particularly in more challenging, fluctuating noise environments. Combined with the optimized tactile vocoder system, these enhancements helped the researchers achieve better performance in terms of noise reduction and tactile speech perception.

# Chapter 3

# Work Plan

Our thesis work is structured into three prominent phases which are Pre-thesis 1, Pre-thesis 2, and Defence. The first phase of Pre-thesis 1 took four months. During this period, we selected the team member, supervisor, and topic for our research and completed the necessary courses and books that were provided to us by our supervisor. Successful confirmation of the topic was followed by the registration of the topic name and thesis abstract. Then, after skimming multiple papers, we identified those that matched our topic. We did an extensive review of the selected papers, summarized them into a literature review, and submitted our findings in the Pre-thesis 1 report. The next stage, that is, Pre-thesis 2, is also most likely to last four months. During this phase, our goals are to collect datasets, analyze the datasets, build a baseline model, and write the Pre-thesis 2 poster and report. The Defense phase is again four months, which will cover the development of an advanced model, performance analysis, completion of the thesis paper, and preparation for the final defense presentation. The following chart shows an approximate step-by-step plan we intend to follow in the future to efficiently conduct our research.

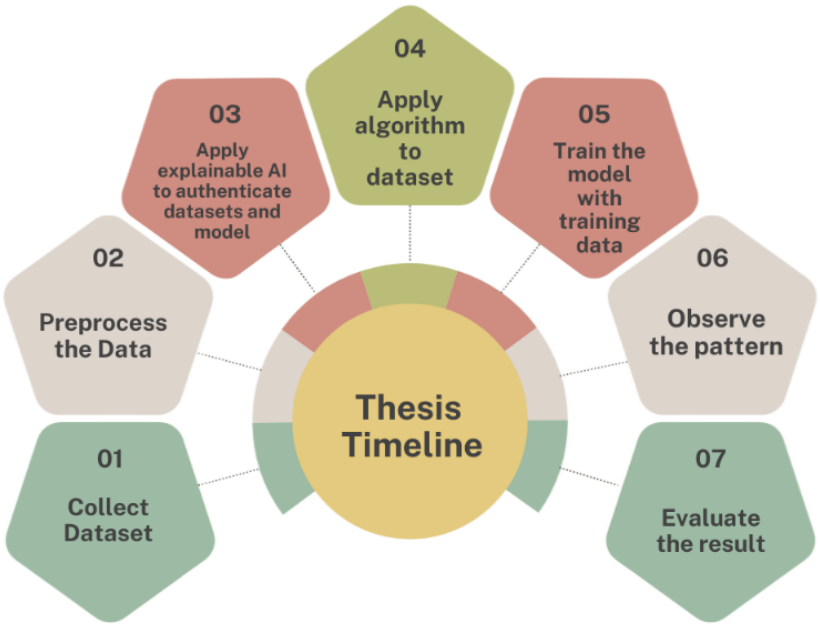


Figure 3.1: Thesis timeline showing main phases.

# Chapter 4

# Methodology

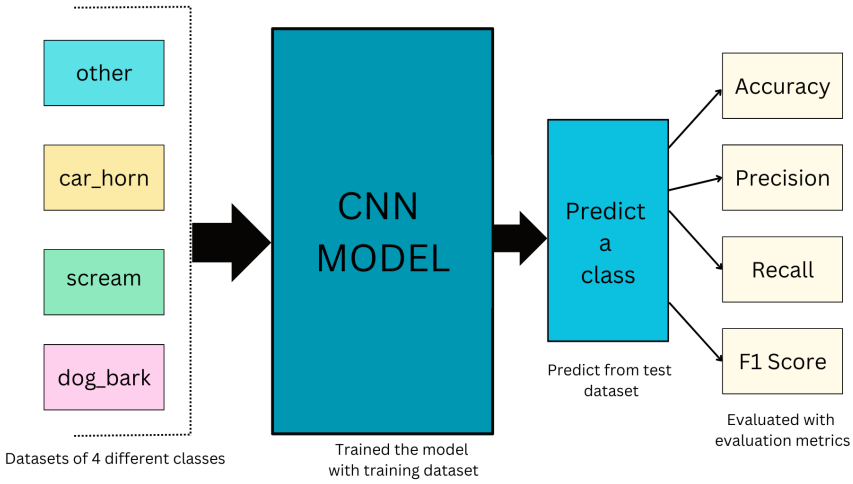


Figure 4.1: Methodology Flow Diagram

figure 4.1 represents the overall methodology of our work which started with data collection & process. We defined 3 different categories of audios on which we will train our model. Those three categories are stated as our 3 different classes: car_horn, scream and dog_bark. Any other sound will be defined as noise and go to the category named other. Which divided our total dataset into four categories. The dataset containing four classes was fed to our CNN model for training. After the training process, test data were passed to check the accuracy of the model and other evaluation matrices were implemented such as precision, recall, and F1 score for the final evaluation.

# Chapter 5

# Dataset

## 5.1 Analysis of Dataset

We worked on audio datasets for recognizing car horns, screams and dog barks. All of them are collected from external sources. There are a total of 8061 audio datasets from which the total dataset is divided into 8061 rows and 4 columns. Columns contain filename, duration, class and target. There are a total of 4 classes which are 'car_horn', 'scream', 'dog_bark' and 'other'. Any kind of noise or environmental sound that is not a car horn or scream or dog bark falls under 'other' class. The target column contains the numeric value against each class. The 'car_horn' class represented 1, 'scream' represented 2, 'dog_bark' represented 3 and 'other' represented 0. The length of each audio file is set to 3 seconds to ensure the fixed input dimension so that the model can process all the data simultaneously.

### 5.1.1 Class Based Analysis

For the portion of the car horn detection dataset, we collected them from three different sources - two different Kaggle datasets and one free audio website platform named 'Freesound'. From 'Freesound', we extracted 998 audio files where 838 contain car horns & 160 contain environmental noise. From the first Kaggle dataset 1080 audio files were extracted containing car horns. Lastly, another Kaggle dataset offered 1920 files that are filled with different types of noises. Thus our total audio files for car horn detection reached 3998. All the general noise and environmental noise fell under 'other' class. For the detection of scream we used a dataset from kaggle that provided a total of 2000 audio files after preprocessing. Both clean screams and noise-mixed screams were included in these audio files. Approximately 27% of the scream dataset was clean scream. For the dog_bark class, another 2000 audio files were processed from three different sources - two datasets from Kaggle and one dataset from huggingface website. Here 47.35% of audio files were noise-mixed dog barks and the rest 52.65% audio files were clean dog barks. 1053 files were clean dog bark and 947 files were mixed with noises.

### 5.1.2 Class Distribution

The class distribution represents the number of samples in each class. We visualized this class distribution for our four different classes. Following is the figure that

represents class distribution:

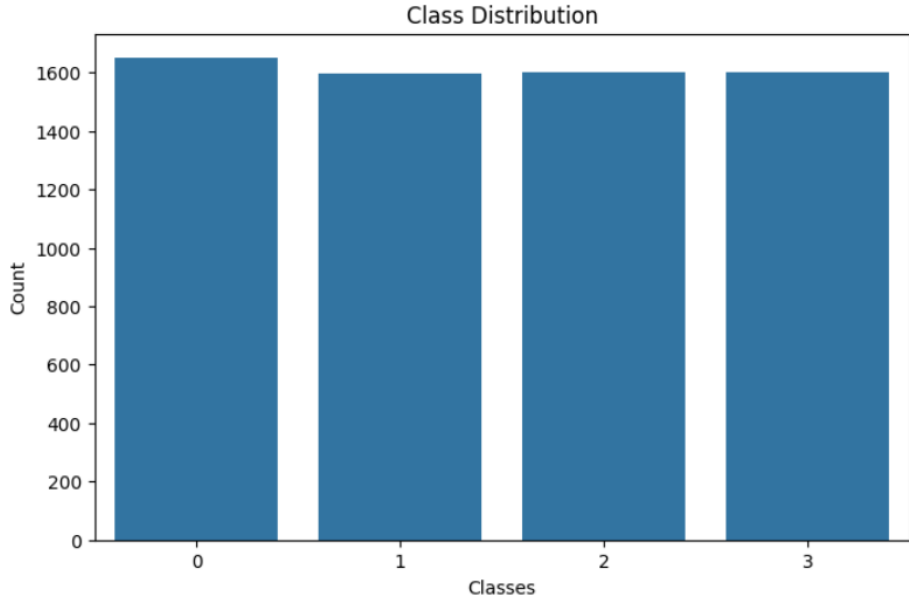


Figure 5.1: Class Distribution of four classes

Here the x-axis represents the target value of each class and the y-axis represents the sample present in the dataset which is not the literal value but rather converted to a range from 0 to 1600. 0 represents the 'other' class, 1 represents the 'car_horn' class, 2 represents the 'scream' class and 3 represents the 'dog_bark' class. As we have supplied an equal number of sample data for each detection, the bar chart represents an equal distribution among them. Noise data('other' class) was slightly higher than the other three classes as we supplied more than 2000 audios for this section. This distribution ensures that there is a perfect balance among classes and there is no underrepresented class in this dataset.

### 5.1.3 Mean & Variance Intensity

Mean & variance intensity analysis helps to understand the energy distribution of a dataset. Mean intensity represents the average loudness or the average energy of an audio. Variance intensity represents the fluctuation of energy or the fluctuation of loudness over time. We plotted the mean intensity and variance intensity for four different classes of the dataset which is as follows:
From the figure 5.2 we can see that the 'other'(0) class holds the highest range of mean intensity which is approximately -55 dB to -35 dB. The 2nd highest intensity range is occupied by the 'car_horn'(1) class which is approximately from -65 dB to -40 dB. Thirdly comes the 'dog_bark'(3) class which has a range from -63 dB to -45 dB approximately. Lastly comes the 'scream' dataset which holds the range from -64 dB to -48 dB approximately. Higher mean intensity indicates audio files with higher energy level and lower mean intensity indicates a subdued audio signal. So it can be easily interpreted that the 'other' class has higher or louder audio signals. It happened because this class holds noise of various kinds which may contain very high pitch and bandwidth. We did not bind our 'other' class under any certain noise level. On the other hand, we collected our scream dataset from one single source

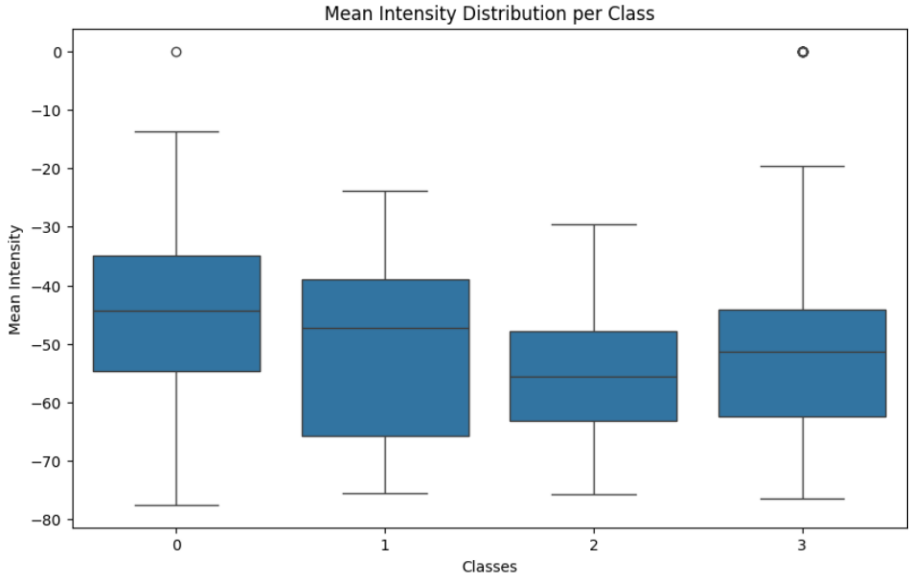


Figure 5.2: Mean Intensity of Four Classes

which contained the same kind of internal quality-based dataset which may lead it to a lower mean intensity.

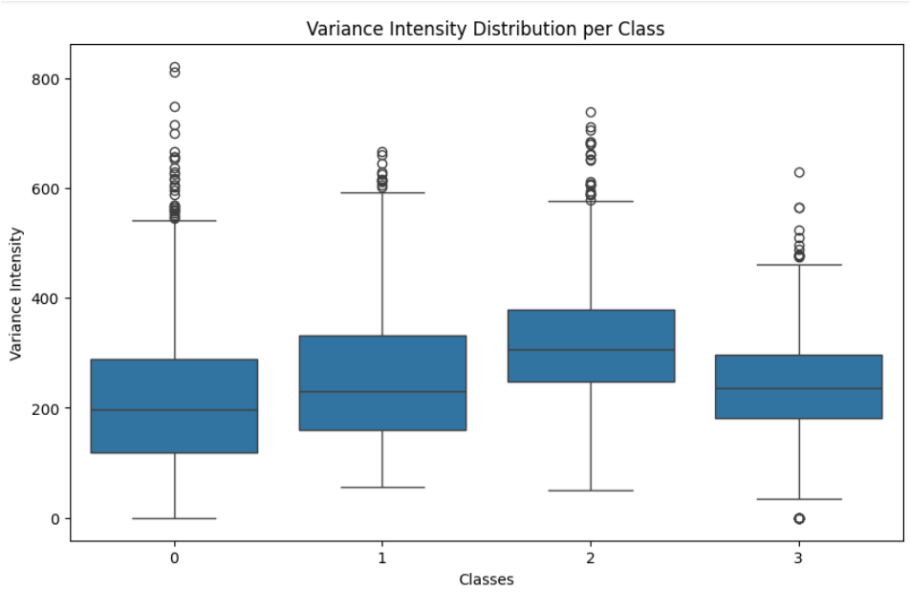


Figure 5.3: Variance Intensity of Four Classes

From the figure 5.3, we got the variance intensity of each class. Higher variance intensity indicates highly fluctuating audio dataset presence and lower variance intensity upholds the opposite. In our dataset 'scream' class has the higher variance intensity which is logical as scream has the tendency of the sudden outburst after a quiet moment. It upholds the range from 250 dB to 350 dB approximately. On the contrary, 'other' class has the lowest intensity ( from 140 dB to 290 dB approximately) which is also reasonable because usually noise holds a continuous form of sound from any environment. The other two classes have medium level variance intensity as expected.

### 5.1.4 Spectrogram Analysis

Spectrograms help represent the distinct characteristics of different types of audio datasets. Since we have four different classes, each of them has a unique spectrogram representing the energy of audio signals across frequencies over time. The plotted spectrograms are as follows:

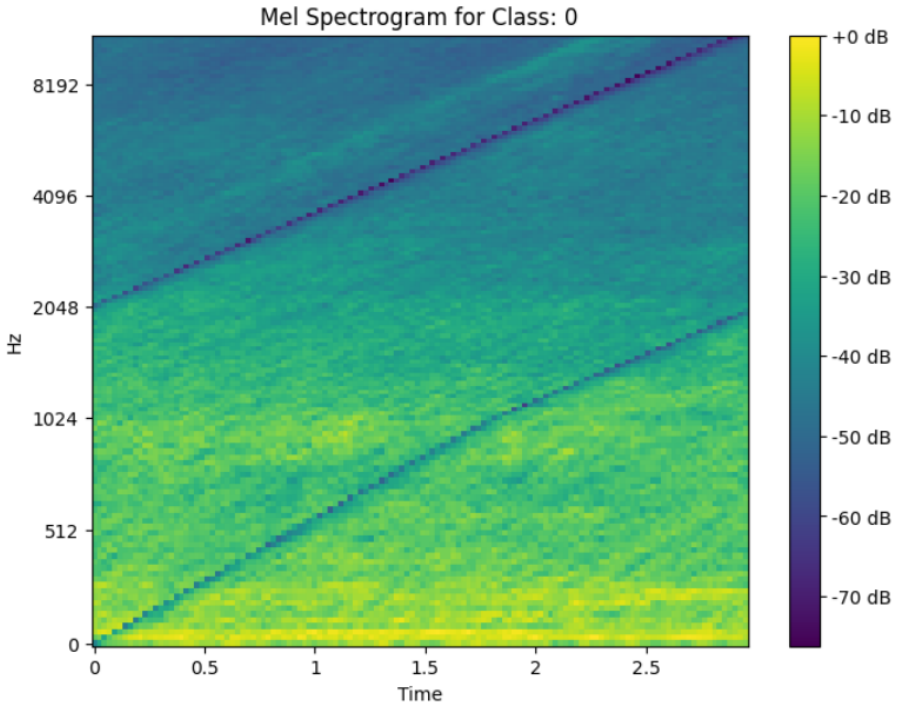


Figure 5.4: Spectrogram of Class 0: Other

Figure 5.4 holds the 'other' class which shows unstructured characteristics because it contains so many kinds of noises that do not maintain any harmonic pattern. It is mainly a fuzzy and diffuse distribution which is broadly spread across frequencies. Randomness of noise makes it unpredictable and shows no constant patterns. There are no sharp peaks or clear lines throughout the spectrogram.

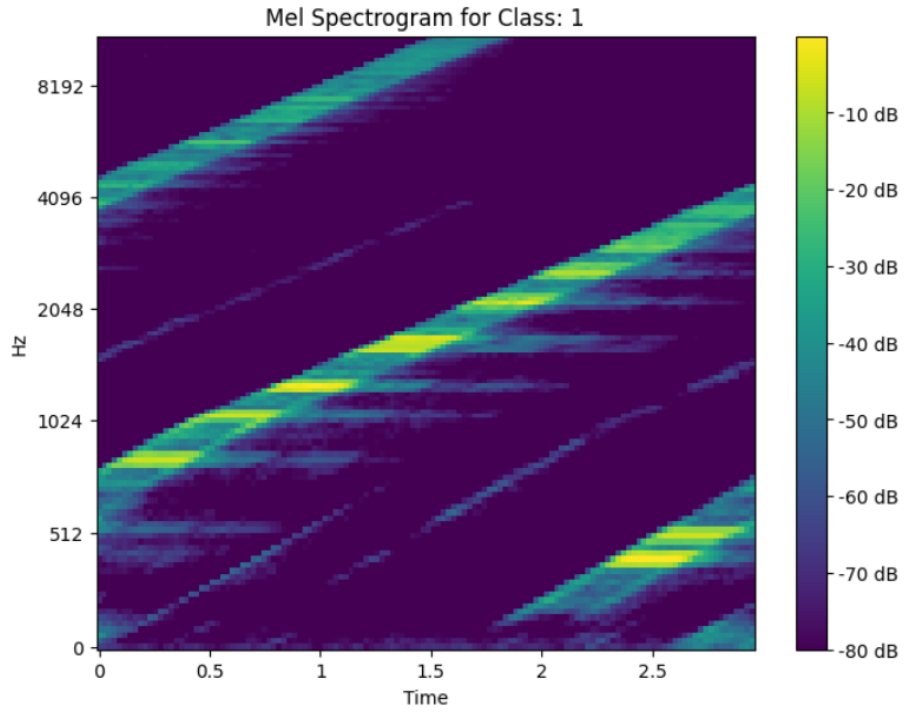


Figure 5.5: Spectrogram of Class 1: Car Horn

In the figure 5.5, we plotted the 'car_horn' class. The brighter part of the spectrogram indicates higher energy across a certain frequency. Unlike noise, it shows a harmonic or periodical pattern throughout the gram. Distinct bands of signals are concentrated in specific frequency bands. No sudden burst of energy is shown which means fewer fluctuations.

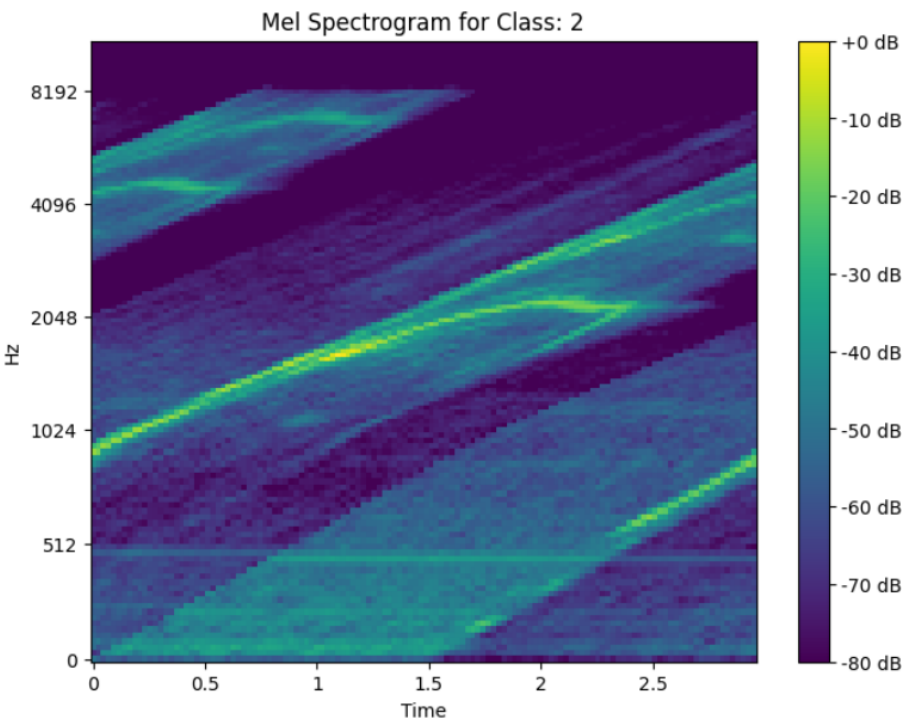


Figure 5.6: Spectrogram of Class 2: Scream

Figure 5.6 shows the 'scream' dataset which holds its common nature with its higher spread of energy in a broader range of frequency. Variability of signal shows high pitch which is likable for scream. Both high and low frequencies show a significant amount of energy. Unlike car horns, it shows irregular patterns in its band.

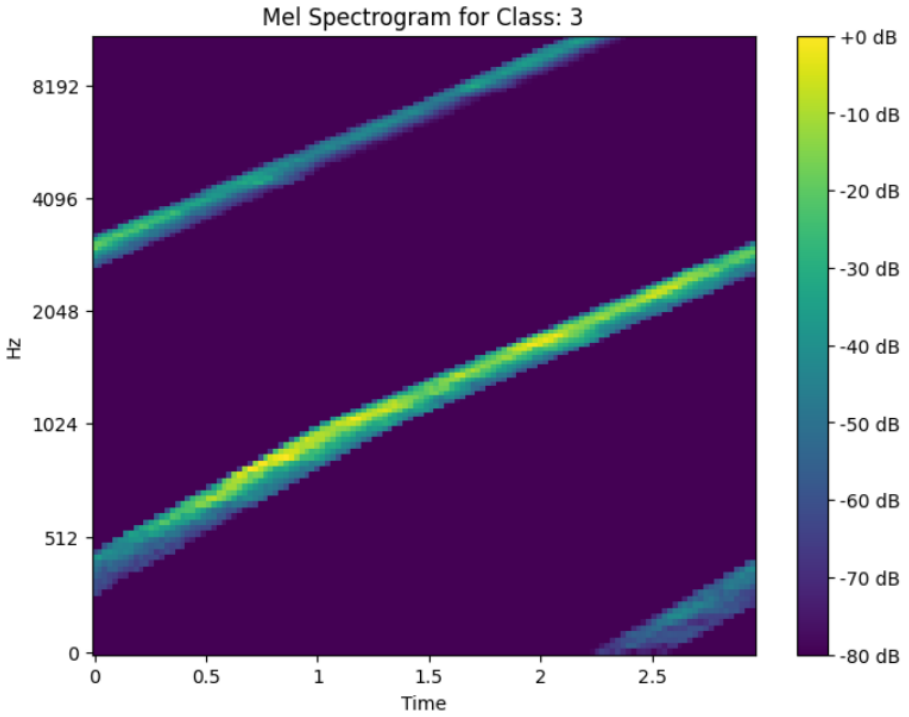


Figure 5.7: Spectrogram of Class 3: Dog Bark

'Dog_bark' dataset is represented in figure 5.7. It is similar to car horn data, but the band of signals is more specified with clearer lines. This is due to the preprocessing of the dog bark dataset, where almost 50% of audio files provided clean dog barks with no background noise. This portion of the dataset holds the highest amount of clean data, so the bands are more specific and periodical.

### 5.1.5 Mean vs Variance Intensity overlaps among Classes

The mean intensity and variance intensity for each class plotted here as a mean vs variance graph which helps to analyze different kinds of features of the dataset. Following figure shows it:
The top-left plotting represents the mean intensity overlaps among classes and bottom-right plotting represents the variance intensity overlaps among classes. From the comparison it can be easily said that variance will be a better feature to distinguish among classes than mean intensity as mean intensity upholds higher overlapping than variance intensity. After doing the scatter plot which is situated at the bottom-left part of the figure, we can declare that maximum overlapping happens between class 1 and 2 which will misguide the separating process. There are comparatively higher chances of getting lower accuracy during predicting class 1 and 2.

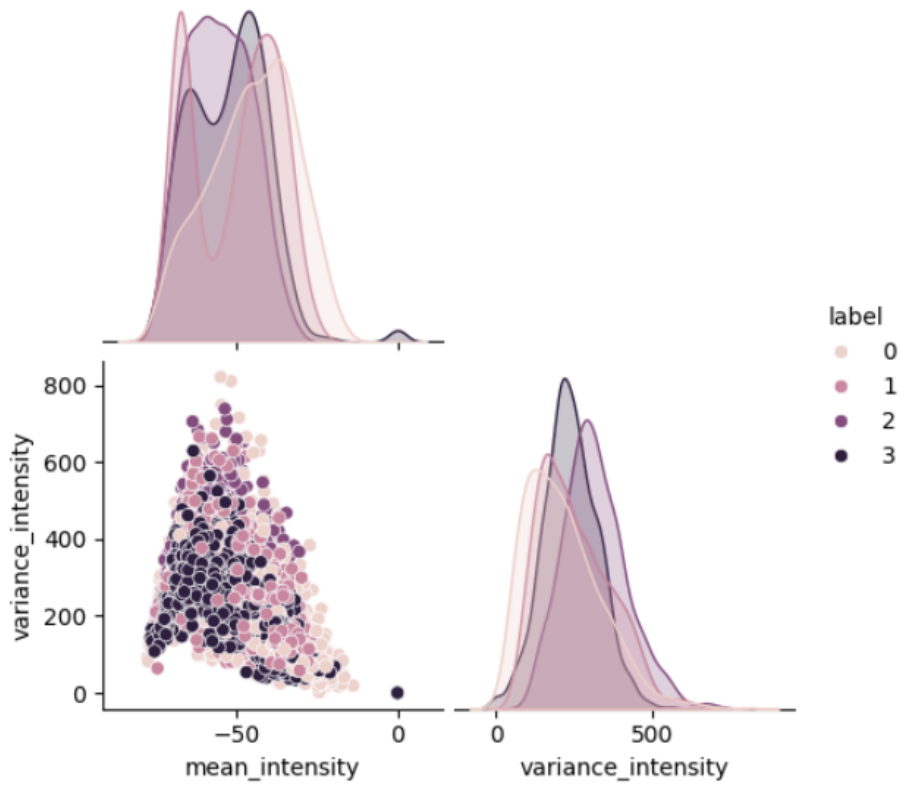


Figure 5.8: Visualization of overlapping features

## 5.2 Dataset Preprocessing

As we worked with four different classes, for each class we collected data from different sources and preprocessed them. After doing individual preprocessing we combined them in a single dataset. After combining, we divided the whole dataset into training and testing parts which are 80% and 20% respectively. validation data was 10%

### 5.2.1 Car Horn and Noise Dataset Preprocessing

For the portion of car horn data, we used three different sources which includes two different Kaggle datasets and one free audio website platform named 'Freesound'. From the website platform, we collected 270 audio files which contained car horn and environmental noise. Then we trimmed these audio files to 3 seconds each and created 918 files. Then we labeled those audio files into three categories - car horn, not car horn and clean car horn. Clean car horn was a part of a car horn which provided minimum background noise. Then we did augmentation on the not car horn part as there were environmental noises which were repetitive making the audio files similar. Thus we created 838 files of car horns & 160 files of environmental noise. Between two datasets from Kaggle, the first one contained 540 audio files which then converted to 1080 audio files containing clear car horns. Here each file length primarily was 1 second. For half of them, we added silent padding both starting and ending of the audio to make it 3 seconds. For the other half, we did lopping to make them 3 seconds long. We also added background noise with a portion of clear car

horn for better learning of the model. Another kaggle dataset contained 2000 audio files from which we collected only 1920 files that are filled with different types of noises and added an extra 80 environmental noise audio files from the previous 160 files of environmental noise collected from 'Freesound' website. Thus we prepared a total 2000 files of noises from this portion. Then we added all the audio files and our total audio files for car horn detection reached 3998. Then we divided them between training and testing where 80% of the dataset were selected for training and 20% of the dataset were for testing. For both parts we made sure to add an equal portion of car horn and noises. Also, we added 30% to 40% clean car horn data in each car horn portion at the beginning. Finally we got two classes from this car horn dataset portion labeled 'car_horn' and 'other' and their target values were 1 and 0 respectively.

### 5.2.2 Scream Dataset Preprocessing

We mainly used one source for the scream dataset which came from a kaggle dataset. Primarily it had 1583 scream audio files and 1583 non-scream or noise audio files. Each of them were 5 seconds long. Firstly we took the 1583 scream audios and trimmed those to make 3 seconds long. Now to balance scream dataset with car horn dataset we randomly took 417 audios from the same scream audios and added noise with those files. These noises were taken from the 1583 non-scream audios. Thus collectively 2000 audio files of scream dataset were prepared and its target value was set 2 under 'scream' class.

### 5.2.3 Dog Bark Dataset Preprocessing

Firstly, we collected 1053 clean dog bark sounds from two different sources; one kaggle dataset and one dataset from huggingface website platform. These audios had different lengths. They were trimmed to 3 seconds. Then with random selection we selected 947 audio files from these 1053 audios and added noise from the third dataset which was collected from kaggle. This dataset was full of different kinds of noises. Thus Prepared total 2000 audio sets and its target value set to 3 under 'dog_bark' class.

# Chapter 6

# Models

## 6.1 Neural Network

Neural networks are a machine learning model that has advanced the field of artificial intelligence by emulating the human brain through processes that imitate how biological neurons work to recognise patterns. These networks consist of layers of nodes or "neurons" that are interconnected and follow a hierarchy respectively of input, hidden and output. Th ese networks work by processing information and learning via mathematical operations of thresholds and weights for activation of nodes.

## 6.2 Deep Learning

Deep learning is a machine learning model that is a subset of neural networks focused on deep architecture that uses multilayered neural networks to learn from data. Deep learning excels with large datasets of high dimensionality for better feature extraction through these greater hidden layers and use it to recognise complex relationships between features and data.

## 6.3 Convolutional Neural Network (CNN)

CNN are specialised deep learning models that are distinguished for their performance with image and spectrograms as they can recognise patterns which are translation invariant and have spatial hierarchy. These are done through the key layers: convolutional layers, pooling layers, and fully connected layers. Audio files are converted from 1D waveforms to 2D spectrograms and then passed into the layers, convolutional layers apply initial filters to detect audio features by creating feature maps through dot products of weights and pixel values, as data progresses through layers more complex patterns are recognised from previous layer's feature maps. We have used 4 convolutional layers that increase filters starting from 32 doubling at each layer and stopping at 256, ReLU function is used after each convolutional layer to introduce non linearity so that complex patterns can be learnt. Pooling layers such as max pooling and average pooling have been used which downsamples parameters by a factor of 2 for greater efficiency and generalization by reducing overfitting. Fully connected layers or dense layers play the pivotal role of classifying

the extracted features based on previous layers to predict the class of the original input. We used 256 neurons in the dense layer to compute a 256 dimension vector that is then passed to the output dense layer with 4 neurons for the 4 classes we have, a softmax function is then applied to them to produce probabilities for the input belonging in each class. Additional layers: dropout layers, flatten layer and batch normalisation layer. Dropout layer is used for reducing overfitting by getting rid of neurons randomly during training to make the model robust. Flatten layer is used to convert multilayered feature maps from pooling layer to one dimensional array before relaying it into the dense layers. Batch Normalization is used to make input of layers more stable by recentering and rescaling, speeding up training.

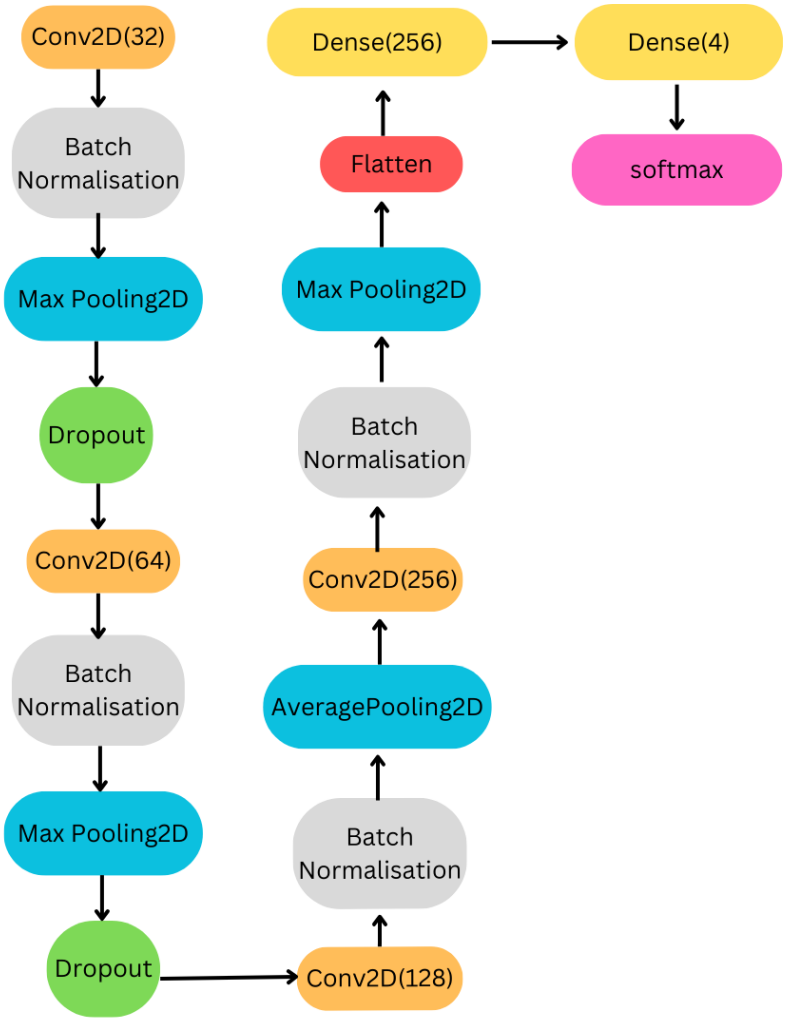


Figure 6.1: CNN Model

# Chapter 7

# Evaluation Metrics

Now we delve into various well known and used evaluation metrics to judge our model's performance on correct sound detection and classification.

**Accuracy**

In sound detection, accuracy means the ratio of correctly detected sounds against the total number of sounds.

Accuracy = Number of Correctly Detected Sounds ÷ Total Number of Sounds in the Dataset

**Precision**

Precision is the measure which shows us the extent our model correctly predicted a particular sound out of the total number of occurrences it detected as that sound where if the precision is high then it will mean fewer false positives in the model's predictions.

Precision = Correctly Predicted Sounds (True Positives) ÷Total Predicted Sounds (True Positives + False Positives)

**Recall**

Recall in sound detection provides the model's ability to retrieve the actual instances of a class predicted properly from the dataset. Here higher recall will ensure that the chances of our model predicting a false negative is fewer or less.

Recall = Number of Correctly Predicted Specific Sounds (True Positives) ÷Total Number of that Specific Sound (True Positives + False Negatives)

### F1 Score

F1-score is the harmonic mean of precision and recall which we use here for judging the quality of prediction of our model and also to minimize errors of precision and recall.

F1 Score = [2 ×Precision ×Recall]÷[Precision + Recall]

# Chapter 8

# Results

With the help of CNN model, we obtained a prediction for each of the audio given which covered four types of sounds such as dog bark, car horn, scream and others. After completing training and testing the model we applied four types of evaluation metrics to measure the performance which included accuracy, precision, recall and f1-score.

Table 1 presents an analysis of the sound detection models CNN on our audio dataset.It showed an accuracy of 0.92 on the dataset which means the model's correct prediction is 92% of the occurrences. Confusion matrices and heat map of the model are also given in figure 8.1 and 8.2 below clearly showing the overall improved performance of the model on our audio dataset.

The performance of our model was improved by fine tuning their hyperparameters. The configuration variables and their values that we used to fine tune are given in table 2. We experimented with the number of layers, activation function , dropout rate rate, patience, epoch number, regularizer repeatedly for a better performance on our output. In figure 8.3, we see how up to the 4th convolution layer the performance of our model increased and in the 5th layer it decreased significantly thus we used 4 convolution layers with 2 dense layers in our model training for better performance.

| **Target** | **Precision** | **Recall** | **F1 Score** |
|---|---|---|---|
| Other | 0.84 | 0.88 | 0.86 |
| Car Horn | 0.94 | 0.93 | 0.94 |
| Scream | 0.96 | 0.94 | 0.95 |
| Dog Bark | 0.93 | 0.92 | 0.93 |

Table 1: Performance overview of CNN for sound detection on the Dataset

| Hyperparameter | Configuration |
|---|---|
| Learning Rate | 0.005 |
| Batch Size | 64 |
| Number of Epoch | 50 |
| Optimizer | Adam |
| Sequence Length | 128*128*1 (input size) |
| Patience | 5 (Early Stopping Callback) |

Table 2: Hyperparameter Tuning on CNN model

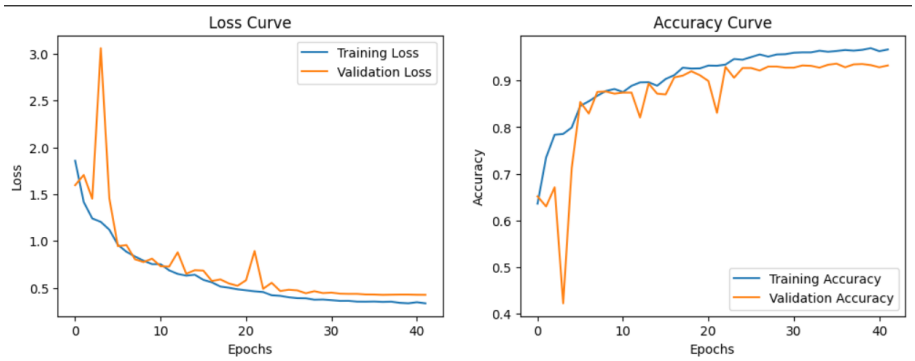


Figure 8.1: Graph of Training and Validation Data loss and accuracy

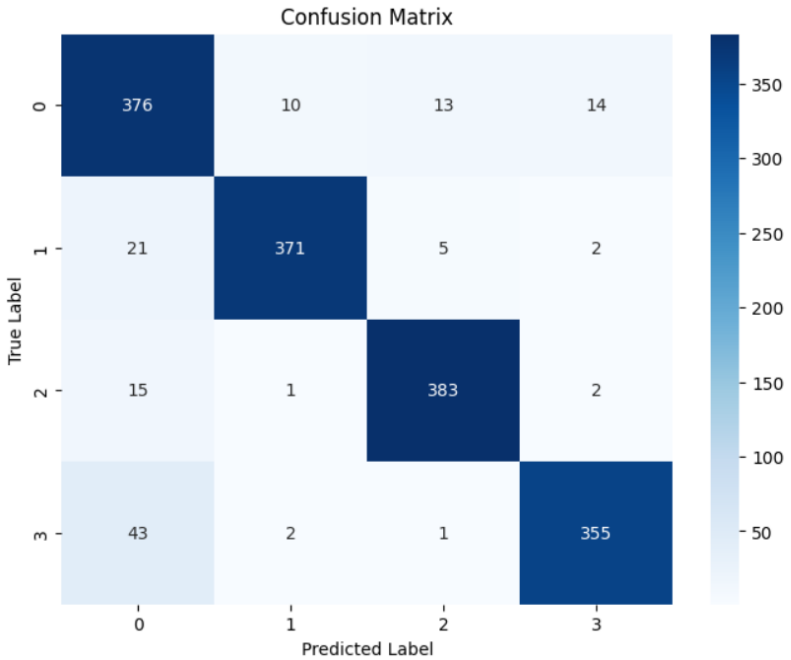


Figure 8.2: Heat map representation of the sound detection confusion matrix

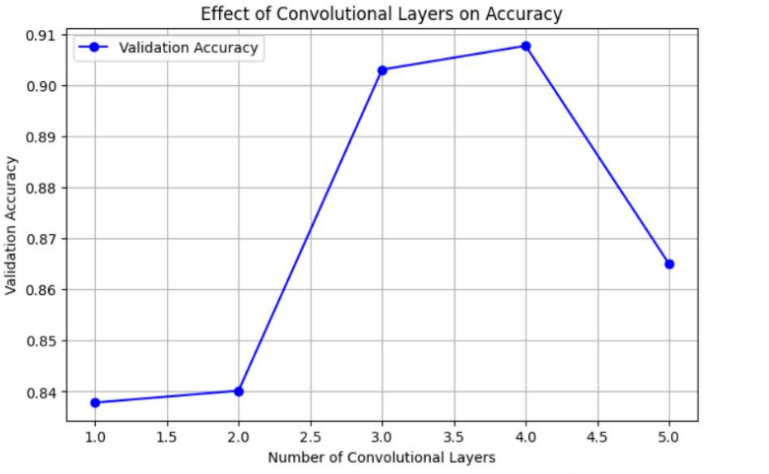


Figure 8.3: Effect of Convolution Layers on Performance

# Chapter 9

# Conclusion

Our research intent is to develop a system that recognizes the source location of specific sounds and gives real-time visual and tactile feedback in a wearable device. This will help hearing-impaired people to be aware of their surroundings because they will be able to know if something needs their attention even if it is out of their sight. Hearing-impaired people can communicate with the help of lip reading and sign language but, for this, they rely on sight. Our proposed system can help them navigate through situations where sight is not involved. For example, if someone calls them from behind they will not only get tactile feedback (vibration) that they have been called but also see from a device what direction the call is coming from. In order to achieve this we used a reliable and efficient model to detect specific words or sounds which is the CNN model and give feedback to the user. We expect our proposed system to be a cost-effective, sensible solution for hearing-impaired people by leveraging modern-day NLP and ML to accurately transcribe, classify, and interpret sounds in real-time, enhancing accessibility and communication in daily life.

# Bibliography


[1] A. Saxena and A. Y. Ng, "Learning sound location from a single microphone," in *2009 IEEE International Conference on Robotics and Automation*, 2009, pp. 1737–1742. DOI: 10.1109/ROBOT.2009.5152861.

[2] H. Lim, I.-C. Yoo, Y. Cho, and D. Yook, "Speaker localization in noisy environments using steered response voice power," *IEEE Transactions on Consumer Electronics*, vol. 61, no. 1, pp. 112–118, 2015. DOI: 10.1109/TCE.2015.7064118.

[3] H. Archana, C. Chellaram, and M. Rajalakshmi, "Obstacle detection for visually impaired patients," in *2014 International Conference on Science Engineering and Management Research (ICSEMR)*, 2014, pp. 1–3. DOI: 10.1109/ICSEMR.2014.7043565.

[4] K. Audhkhasi, A. Rosenberg, A. Sethy, B. Ramabhadran, and B. Kingsbury, "End-to-end asr-free keyword search from speech," *IEEE Journal of Selected Topics in Signal Processing*, vol. 11, no. 8, pp. 1351–1359, 2017. DOI: 10.1109/JSTSP.2017.2759726.

[5] P. Vecchiotti, E. Principi, S. Squartini, and F. Piazza, "Deep neural networks for joint voice activity detection and speaker localization," in *2018 26th European Signal Processing Conference (EUSIPCO)*, 2018, pp. 1567–1571. DOI: 10.23919/EUSIPCO.2018.8553461.

[6] M. S. Mahamud and M. S. R. Zishan, "Watch it: An assistive device for deaf and hearing impaired," in *2017 4th International Conference on Advances in Electrical Engineering (ICAEE)*, 2017, pp. 556–560. DOI: 10.1109/ICAEE.2017.8255418.

[7] A. H. Michaely, X. Zhang, G. Simko, C. Parada, and P. Aleksic, "Keyword spotting for google assistant using contextual speech recognition," in *2017 IEEE Automatic Speech Recognition and Understanding Workshop (ASRU)*, 2017, pp. 272–278. DOI: 10.1109/ASRU.2017.8268946.

[8] M. Yağanoğlu and C. Köse, "Real-time detection of important sounds with a wearable vibration based device for hearing-impaired people," *Electronics*, vol. 7, no. 4, 2018, ISSN: 2079-9292. DOI: 10.3390/electronics7040050. [Online]. Available: https://www.mdpi.com/2079-9292/7/4/50.

[9] H. Seki, T. Hori, S. Watanabe, J. Le Roux, and J. R. Hershey, "A purely end-to-end system for multi-speaker speech recognition," in *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, I. Gurevych and Y. Miyao, Eds., Melbourne, Australia: Association for Computational Linguistics, Jul. 2018, pp. 2620–2630. DOI: 10.18653/v1/P18-1244. [Online]. Available: https://aclanthology.org/P18-1244.

[10] M. D. Fletcher, A. Hadeedi, T. Goehring, *et al.*, "Electro-haptic enhancement of speech-in-noise performance in cochlear implant users," *Scientific Reports*, vol. 9, p. 11 428, Aug. 2019. DOI: 10.1038/s41598-019-47718-z. [Online]. Available: https://doi.org/10.1038/s41598-019-47718-z.

[11] X. Chang, W. Zhang, Y. Qian, J. L. Roux, and S. Watanabe, "Mimo-speech: End-to-end multi-channel multi-speaker speech recognition," in *2019 IEEE Automatic Speech Recognition and Understanding Workshop (ASRU)*, 2019, pp. 237–244. DOI: 10.1109/ASRU46091.2019.9003986.

[12] S. Shen, D. Chen, Y.-L. Wei, Z. Yang, and R. R. Choudhury, "Voice localization using nearby wall reflections," in *Proceedings of the 26th Annual International Conference on Mobile Computing and Networking*, ser. MobiCom '20, London, United Kingdom: Association for Computing Machinery, 2020, ISBN: 9781450370851. DOI: 10.1145/3372224.3380884. [Online]. Available: https://doi.org/10.1145/3372224.3380884.

[13] M. D. Fletcher, H. Song, and S. W. Perry, "Electro-haptic stimulation enhances speech recognition in spatially separated noise for cochlear implant users," *Scientific Reports*, vol. 10, p. 12 723, Jul. 2020. DOI: 10.1038/s41598-020-69697-2. [Online]. Available: https://doi.org/10.1038/s41598-020-69697-2.

[14] C. Kim, D. Gowda, D. Lee, *et al.*, "A review of on-device fully neural end-to-end automatic speech recognition algorithms," in *2020 54th Asilomar Conference on Signals, Systems, and Computers*, 2020, pp. 277–283. DOI: 10.1109/IEEECONF51394.2020.9443456.

[15] M. H. Korayem, S. Azargoshasb, A. H. Korayem, and S. Tabibian, "Design and implementation of the voice command recognition and the sound source localization system for human–robot interaction," *Robotica*, vol. 39, no. 10, pp. 1779–1790, 2021. DOI: 10.1017/S0263574720001496.

[16] M. D. Fletcher, J. Zgheib, and S. W. Perry, "Sensitivity to haptic sound-localization cues at different body locations," *Sensors*, vol. 21, no. 11, 2021, ISSN: 1424-8220. DOI: 10.3390/s21113770. [Online]. Available: https://www.mdpi.com/1424-8220/21/11/3770.

[17] H. Aldarmaki, A. Ullah, S. Ram, and N. Zaki, "Unsupervised automatic speech recognition: A review," *Speech Communication*, vol. 139, pp. 76–91, 2022, ISSN: 0167-6393. DOI: https://doi.org/10.1016/j.specom.2022.02.005. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0167639322000292.

[18] S. Chen, R. Tan, Z. Wang, X. Tong, and K. Li, "Voicemap: Autonomous mapping of microphone array for voice localization," *IEEE Internet of Things Journal*, vol. 11, no. 2, pp. 2909–2923, 2024. DOI: 10.1109/JIOT.2023.3294937.

[19] M. D. Fletcher, C. A. Verschuur, and S. W. Perry, "Improving speech perception for hearing-impaired listeners using audio-to-tactile sensory substitution with multiple frequency channels," *Scientific Reports*, vol. 13, p. 13 336, Aug. 2023, Received 03 March 2023, Accepted 11 August 2023, Published 16 August 2023. DOI: 10.1038/s41598-023-40509-7. [Online]. Available: https://doi.org/10.1038/s41598-023-40509-7.

[20] M. D. Fletcher, S. W. Perry, I. Thoidis, *et al.*, "Improved tactile speech robustness to background noise with a dual-path recurrent neural network noise-reduction method," *Scientific Reports*, vol. 14, p. 7357, Mar. 2024. DOI: 10.1038/s41598-024-57312-7. [Online]. Available: https://doi.org/10.1038/s41598-024-57312-7.