JOURNAL OF BASIC SCIENCE AND ENGINEERING

Detection and Classification of Diverse Listening Conditions for Hearing-Impaired Individuals using RNN Model and FIR Filter

Sunilkumar M. Hattaraki [a], Shankarayya G. Kambalimath [b]

[a] Department of Electronics and Communication
Engineering, B.L.D.E.A's V. P. Dr. P. G. Halakatti
College of Engineering and Technology,
Vijayapura-586103, Karnataka, India.
Visvesveraya Technological University, Belagavi-
590018. Karnataka, India.

[b] Department of Electronics and Communication
Engineering,
Basaveshwar Engineering College,
Bagalkote, Karnataka, India.
Visvesveraya Technological University, Belagavi-
590018. Karnataka, India.

DOI: https://doie.org/10.0802/Jbse.2024984437

Abstract

Machine learning has numerous applications in audio-signal classification. It helps to find and sort different kinds of sounds, such as talking, music, and noise, from the world around us. Before applying machine learning to classify audio signals, the audio is first converted into a format that the computer can understand. Sound is shown using methods such as pictures of sound waves, special numbers called Mel-frequency Cepstral coefficients (MFCC), a method of predicting sound patterns called linear predictive coding, and breaking down sound into tiny parts using wavelet decomposition. Once the audio has been formatted appropriately, it can be used as input for a machine learning (ML) model intended for classification. This paper introduces an approach utilizing a Recurrent Neural Network (RNN) model integrated with Finite Impulse Response (FIR) filtering. This combination aims to effectively detect and classify various listening conditions, particularly tailored to individuals with hearing impairment. The pre-trained RNN model accurately categorizes audio, while dynamic FIR filtering enhances audio quality based on the predicted environment, tailored to address the needs of hearing-impaired individuals. This model enables the detection and classification of diverse listening conditions with a training accuracy of 98.50% and a testing accuracy of 94.97%, offering personalized filtering to enhance auditory experiences for hearing-impaired individuals.

Keywords: Detection, Classification, Environmental Sound, Machine Learning Model (ML), Recurrent Neural Network (RNN), FIR filter and Hearing-Impaired Individuals.

1. Introduction

There are numerous types of audio classification, such as speech, music, acoustic data, natural language, and ambient sounds. The ML-model excels in audio classification because it identifies the unique properties of audio samples and uses that knowledge to classify them into multiple categories. A sound signal is used as input for the audio classification process, where its features are retrieved and allocated to the appropriate output category [2-4].

Hearing impairment, a prevalent sensory disability affecting millions of people worldwide, significantly affects individuals' ability to perceive and interact with their acoustic environment. Recognizing the importance of addressing the unique auditory challenges faced by hearing-impaired individuals, this study introduces a pioneering approach to detect and classify diverse listening conditions using advanced technologies. By leveraging the capabilities of Recurrent Neural Network (RNN) models and finite impulse response (FIR) filtering, this research aims to provide tailored solutions that enhance auditory experiences.

The sensory landscape of hearing-impaired individuals is marked by various environmental factors, each presenting distinct acoustic characteristics that influence their ability to comprehend and engage with auditory stimuli. From quiet settings to bustling urban environments, the diversity of listening conditions poses significant challenges for individuals with hearing impairments, impacting their communication, social interaction, and overall well-being. Traditional assistive devices and interventions, although effective to some extent, often fail to adequately address the needs arising from this complexity.

The use of assessment measures is crucial for evaluating classification systems like deep learning. These metrics provide a complete set of indicators for measuring the performance of deep learning models. These are crucial in measuring how effectively a model absorbs information from its training data, as well as identifying areas for development to maximize its efficacy[1],

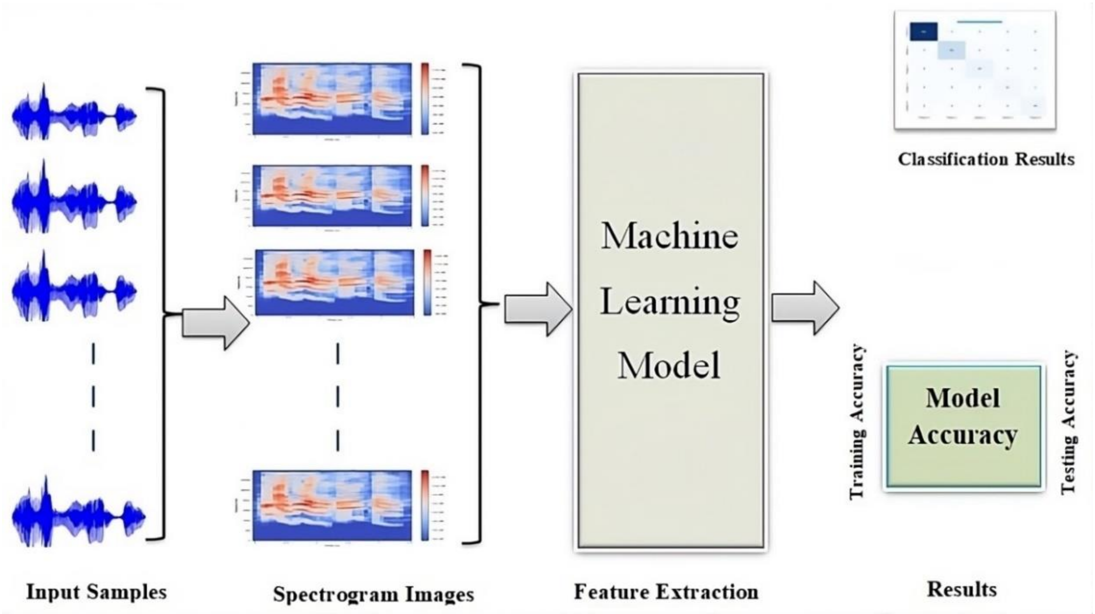


Fig.1: Machine learning model for Environmental Sound Classification (ESC)

Fig.1 shows illustration of a machine learning model designed for ESC. To create a robust audio detection and classification system that takes advantage of the diverse audio data available in the NOIZEUS and Urbansound8K datasets [15][16]. The process begins with meticulous data pre-processing, which involves carefully curating and partitioning audio files into distinct sets for training and testing. The audio files were then transformed into spectrogram images, which allowed us to capture the temporal and frequency characteristics of the audio signals effectively.

Furthermore, visualization approaches such as confusion matrices are used to look further into the model's behaviour, discover patterns of misclassification, and iteratively improve our methodology.

2. Related Work

Veena et al. [5] developed a sound classification system for hearing-impaired individuals (SCSHIP). It combines IoT and Machine Learning to analyze live audio and send immediate alerts. This system fills a gap in connectivity for the hearing-impaired, promising to improve their sense of surroundings and societal integration [25].

Khamparia et al. [6] used deep-learning networks to classify environmental sounds using spectrogram images. They trained CNN and TDSN models on datasets. This approach shows encouraging prospects for crafting proficient sound classification and recognition systems, shedding light on the utilization of spectrogram-based techniques in the realm of deep learning.

Su et al. [7] addressed the limitations of existing deep learning models for ESC by proposing two combined features to enhance representation. Then CNN model is introduced to leverage these aggregated features, significantly improving the performance. Experimentation revealed that our approach achieves a classification accuracy of 97.2% on UrbanSound8K datasets, surpassing previous models and demonstrating its efficacy in environment sound categorization tasks.

Zhao et al. [8] created a deep-learning method that reduces room reverberation and background noise. They improved the voice clarity of hearing-impaired individuals in loud surroundings by training a DNN to estimate an optimum ratio mask. The algorithm also helped normal-hearing listeners by approaching or matching the voice clarity of raw audio. This study represents a big step forward in applying deep learning to improve speech comprehension for the hearing impaired in everyday situations [26].

Mushtaq et al. [9] proposed combining DCNN for ESC. They also used regularization and data augmentation to boost performance. They employed log-Mel features on supplemented data, achieved the best accuracies: 94.94%. These data show how effective their strategy is in addressing sound categorization issues.

Mushtaq et al. [10] developed a new method to classify environmental sounds using Convolutional Neural Networks (CNN) with smart tricks to improve the data, based on Mel spectrograms. They tried different CNN models, such as those with seven layers and nine layers built from scratch, as well as some techniques where they used parts of already trained models, freezing the early layers, and then fine-tuning them to fit their task. Instead of simply changing the images as usual, they came up with ways to improve the audio clips. The results showed that their approach worked well, with high accuracy rates on all datasets. Models such as ResNet-152 and DenseNet-161 performed exceptionally, with accuracy 99.49%. These findings indicate a significant step forward in the accurate ESC.

Hadi et al. [11] advocated the use of machine learning to detect different types of urban noise. They used four supervised algorithms on a dataset They employed MFCC to extract information from the recordings. The results revealed that all algorithms classified noise with accuracies ranging from 95% to 100%. It is vital to note that the recorded noise levels surpassed the World Health Organization's recommendations, potentially endangering people's health.

Toffa et al. [12] proposed a unique method known as a local binary pattern (LBP) with audio features. They adapted LBP, which is generally used for image identification. When compared to standard features, they discovered that LBP characteristics outperformed them. Although not the latest technology, this method offers speed advantages over CNN methods and is preferable with limited data or computing resources.

Demir et al. [13] introduced a unique method for categorizing environmental noises based on deep features extracted from fully connected layers of a Convolutional Neural Network. They trained the model from start to finish with spectrogram images, and then combined these fully connected layers to form a feature vector. When they tested their approach, they discovered that it was quite effective with accuracies of 96.23%.

Z. Chi et al. [14] presented a novel deep convolutional neural network. This network employs mixed spectrograms, notably Log-Mel and Log-Gammatone spectrograms, to provide more detailed information than utilizing only one type of spectrogram. Their network was made up of blocks with three layers for convolution and one layer for pooling to extract key characteristics from the combined spectrograms. They employed tiny filters in each convolution layer to keep the network deep but not overly complicated, using average pooling to preserve the information. When they tested their network on the datasets, they got classification accuracies of 83.8%. This suggests that their strategy is effective for categorizing environmental sounds.

The reviewed papers collectively demonstrate remarkable progress in environmental sound classification (ESC). These studies introduced innovative approaches such as IoT integration, data augmentation, and deep feature extraction from CNNs, significantly improving the classification accuracy and system robustness. Despite these advancements, challenges persist in integrating Finite Impulse Response (FIR) filtering with recurrent neural networks (RNNs) for ESC tasks. Further research is needed to explore this integration and its potential to enhance the feature extraction and classification accuracy in noisy environments.

Although recent advances in ESC have yielded significant improvements, challenges remain in integrating FIR filtering with RNNs. This integration can enhance feature extraction and classification accuracy by capturing temporal dependencies and filtering out noise from audio signals. However, the integration of FIR filtering with RNNs requires further exploration to address existing gaps and improve the ESC performance in dynamic environments.

3. Environmental Sound Detection and Classification Using Integrated RNN Model with FIR Filter

This section introduces Environmental Sound Detection and Classification using an Integrated RNN Model with FIR Filter, a method proficient in categorizing environmental sounds accurately. By integrating the RNN model and FIR filter, it offers a comprehensive approach to analyzing environmental sounds, leveraging the RNN's sequential data processing for capturing temporal patterns and the FIR filter's feature extraction and noise reduction capabilities. This combined approach enhances classification precision by focusing on crucial audio characteristics, resulting in a robust and efficient environmental sound classification system that can handle diverse contexts effectively.

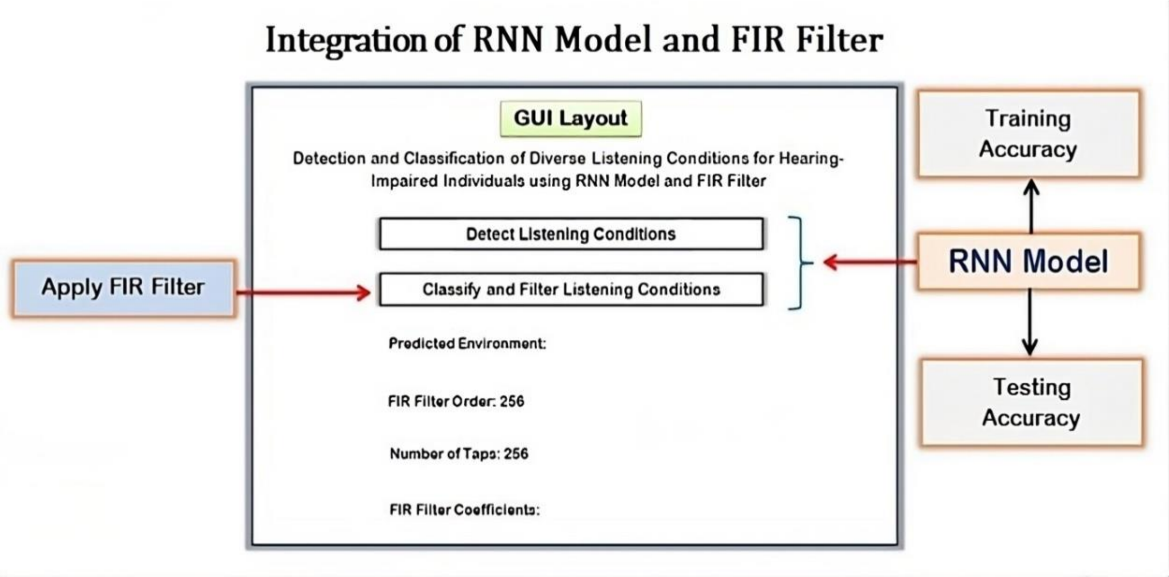

Integration of RNN Model and FIR Filter
GUI Layout
Detection and Classification of Diverse Listening Conditions for Hearing-Impaired Individuals using RNN Model and FIR Filter
Detect Listening Conditions
Classify and Filter Listening Conditions
Predicted Environment:
FIR Filter Order: 256
Number of Taps: 256
FIR Filter Coefficients:
Apply FIR Filter
Training Accuracy
RNN Model
Testing Accuracy

Fig.2: Environmental Sound Detection and Classification (ESC) using Integrated RNN Model with FIR Filter.

Fig. 2 illustrates the process of ESC using an RNN Model with FIR filter. The figure depicts the architecture of the integrated model, including how the sound data are inputted, processed through the RNN layers, and filtered using FIR filters to improve classification accuracy.

This model serves as a versatile tool for detecting and classifying diverse listening conditions tailored to individuals with hearing impairment. It imports the essential libraries for audio and machine learning. The data directories for various environments were defined to access the audio files corresponding to each condition. Furthermore, a pre-trained RNN model was loaded for accurate audio classification. Key functions are implemented to calculate the FIR filter coefficients, apply filters to audio data, and display filter information.

The FIR filter coefficients were calculated based on the sample rate, target environment, and filter order. The filter order was calculated as follows:

$$\text{Filter order} = \text{int}\left(\frac{4 * \text{sample rate}}{\text{Desired Transition Width}}\right) \qquad 1$$

The model constructs an intuitive graphical user interface using ipywidgets, enabling users to upload audio files and classify the listening conditions. Upon uploading an audio file and triggering the classification process, the model predicts the environment type using the RNN model and applies an FIR filter accordingly. To enhance flexibility, specific FIR filter coefficients were assigned to each environment type, allowing for tailored filtering based on the predicted conditions. Overall, this model provides a seamless and informative experience for individuals to understand and adapt to varying listening environments, specifically catering to the needs of those with hearing impairment.

3.1 Data Collection

A diverse dataset of audio recordings encompassing various listening environments relevant to hearing-impaired individuals, such as quiet settings, car environment, cocktail environment, restaurant environment, street environment, airport environment, train station environment, group setting environment, reverberant spaces environment, and telephone conversations from the NOIZEUS corpus database and UrbanSound8K dataset, is available on Kaggle [15-16][19].

3.2 Data Pre-processing

MFCCs are derived from a logarithmic transformation of the power spectrum, followed by a linear cosine transformation on a scale called the mel frequency [2] [20-21][23]. The process of computing the MFCCs involves several steps. First, the signal was enhanced to produce high-frequency components. Then, it is broken into overlapping pieces, each treated with a window function to avoid mixing the frequencies. Next, FFT was used to find the frequency spectrum for each piece. After determining the magnitude of the FFT, it was squared to obtain the power spectrum. This spectrum was then adjusted to the mel-frequency scale by using a special set of filters. The resulting mel-frequency spectrum was converted to a logarithm. Finally, the Discrete Cosine Transform (DCT) was applied to obtain a set number of MFCCs from the logarithm of the mel-frequency spectrum.

3.3 RNN- Model Architecture

Fig. 3 presents a visual representation of the architecture of a RNN model. It may depict the various layers of the RNN, including the input, hidden, and output layers, as well as the connections between these layers. In addition, the figure highlights specific features of the RNN architecture, such as recurrent connections that allow the network to retain information over time. Overall, Fig. 3 provides insight into the internal workings and design of the RNN model, offering a visual guide to its architecture.

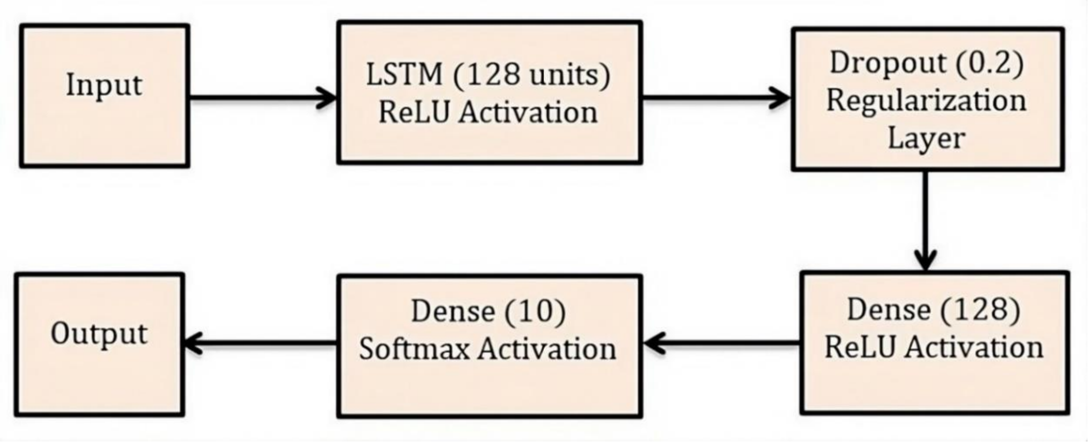

Input
LSTM (128 units) ReLU Activation
Dropout (0.2) Regularization Layer
Dense (128) ReLU Activation
Dense (10) Softmax Activation
Output

Fig.3: RNN-Model Architecture

RNNs learn from sequential data, retaining information over time to capture temporal relationships [22] [27]. They excel in tasks such as audio signal classification and identifying patterns in sound recordings to differentiate between speech, music, and other sounds. RNNs extract features such as frequency and amplitude to understand the content of audio signals, making them valuable for tasks such as audio classification [1][3][5].

RNN is created utilizing the Keras package and a Tensor Flow backend. It begins by creating a sequential model, in which layers are added consecutively. The first layer had 128 units and used a ReLU activation function. LSTMs can handle data sequences, which is beneficial for applications like sorting audio or time-series data. To prevent overfitting, a Dropout layer with a dropout rate of 0.2 is introduced, which turns off 20% of the neurons during training. The network was then made more complicated by adding a Dense layer with 128 units with ReLU activation. Finally, there is another thick layer with ten units and a softmax activation function, which is ideal for categorizing input into one of ten possible groups. This architecture effectively detects patterns in data over time and learns to appropriately classify them [24].

3.3.1. Model Training

In this segment, the defined RNN model was trained using augmented data for 30 epochs, with a batch size of 32. During training, the model's performance was evaluated using validation data. Additionally, the early_stopping callback was employed to monitor the validation loss, allowing training to halt if no improvement was observed for five consecutive epochs, while ensuring that the best-performing weights were retained. This aids in preventing overfitting and improves the generalization capability of the model.

3.3.2 Evaluation and Fine-Tuning

This code segment was used to evaluate the performance of the trained RNN model on the test data. First, a prediction method is applied to the test data using the trained model. This generated the predicted class probabilities for each input sample. Next, the argmax function was used to determine the predicted class labels by selecting the index with the highest probability for each sample along the second axis of the predicted probability array. This resulted in a list of predicted class labels corresponding to the test samples. These predicted labels can then be compared with the true labels to assess the accuracy and performance of the model on the unseen data [2].

Evaluate the essential metrics, including accuracy, precision, recall, and F1-score, to measure the model's capability to accurately identify and categorize various environments. Utilize insights obtained from this evaluation phase to adjust the model parameters and architecture, with the goal of enhancing the overall classification accuracy and resilience.

This methodology can be used to develop a robust and effective system for detecting and classifying diverse listening conditions in hearing-impaired individuals using an RNN model.

Table 1. Dimension and operations of the proposed model.

Layer Type	Output Shape	Parameters
LSTM	(None, 128)	4 * ((input_dim + 1) * 128 + 128 * 128)
Dropout	(None, 128)	None
Dense	(None, 128)	(128 + 1) * 128
Dense	(None, 10)	(128 + 1) * 10

This table 1 outlines the architecture and operations of the proposed model, detailing the output shape and parameters of each layer.

The model begins with a Long Short-Term Memory (LSTM) layer, which outputs a tensor with shape (None, 128). The number of parameters for this layer is calculated using a formula specific to the LSTM layers, accounting for the input dimension and number of units (128 in this case).

Following the LSTM layer, a dropout layer was applied, maintaining the same output shape of (None, 128) while having no trainable parameters. Subsequently, two dense layers were added. The first dense layer produces an output tensor with a shape of (None, 128), with the number of parameters determined by the formula: $(128 + 1) \times 128$. Similarly, the second dense layer generates an output tensor of shape (None, 10), with the number of parameters calculated using the formula $(128 + 1) \times 10$.

Overall, this table provides a clear overview of the model's structure and the computations involved in each layer, helping us to understand the model's architecture and parameterization in detail.

3.4 FIR Filtering

In this section, important functions designed to improve how Finite Impulse Response (FIR) filters work with audio are described. FIR filters are tools used to fine-tune specific aspects of sounds in different listening scenarios.

The "filter order" is a critical factor in FIR filters. This determines the length and complexity of the filter. These functions determine the best settings for the filter by considering factors such as the speed at which the sound was recorded, the part of the sound that needs adjustment, and the length of the filter. By carefully analyzing these factors, these functions ensure that the filter performs optimally in various listening situations.

Furthermore, the filter order affects the effectiveness of the FIR filter when applied to sound. These functions smoothly integrate the FIR filter into sound using predefined settings, taking into account details such as the characteristics of the sound itself and its recording speed. They utilize a specialized function called "scipy.signal.lfilter" to accurately execute this process. Choosing an appropriate filter order enhances the model's ability to accurately analyze and categorize sounds, thereby improving its performance across different listening environments.

The FIR filter was seamlessly applied to the sound within the "apply_fir_filter function. This function considers essential factors, such as sound data, filter settings, and recording speed, to ensure the precise application of the FIR filter. By leveraging the "scipy.signal.lfilter function," it guarantees the accurate integration of the FIR filter into the sound signal. This refinement ultimately enhances the capacity of the model to effectively interpret and categorize sounds across diverse listening contexts.

4. Results and Discussions

The NOIZEUS corpus was created to help researchers compare speech-improvement methods. Inside this database, there are 30 sentences from IEEE spoken by three men and three women. These sentences were mixed with different real-life noises. These noises were obtained from the AURORA database [15-16][19].

The UrbanSound8K dataset, found on Kaggle, contains over 8,000 audio recordings representing diverse urban sounds. Across ten sound classes, ranging from street noise to mechanical sounds, it provides valuable resources for audio classification and urban sound analysis research [17].

Table 2: Dataset Distribution for Environmental Sound Classification - Training and Testing Files.

S.No.	Environment Type	Training (Number of Audio files)	Testing (Number of Audio files)
1	Quiet Environment	2,000	400
2	Car Noise Environment	100	10
3	Cocktail Environment	100	10
4	Restaurant Environment	100	10
5	Street Environment	100	10
6	Airport Environment	100	10
7	Train Station Environment	100	10
8	Group Setting Environment	140	14
9	Reverberant Spaces Environment	50	05
10	Telephone Conversations	100	10
Total No. of Audio Files		**2,890**	**489**

The table 2 provides a brief overview of the various environmental kinds, as well as the number of audio files accessible for training and testing. It describes a variety of circumstances, including quiet environments, car noise, cocktail environment, restaurant environment, street environment, airport environment, train station environment, group settings environment, reverberant spaces environment, and telephone talks. Each environment type is identified by the quantity of audio recordings available for training and testing, allowing audio processing algorithms or models to be evaluated and developed. There are a total of 2,890 audio files accessible for training and 489 for testing, allowing for thorough investigation and evaluation across a wide range of acoustic settings.

Table 3: Performance Metrics for Environmental Sound Classification

Environment Type	Precision	Recall	F1-Score	Support
Quiet Environment	1.00	1.00	1.00	401

Car Noise Environment	0.96	0.96	0.96	24
Cocktail Noise Environment	1.00	1.00	1.00	14
Restaurant Noise Environment	1.00	0.75	0.86	16
Street Noise Environment	0.94	0.76	0.84	21
Airport Noise Environment	0.64	0.86	0.73	21
Train Station Noise Environment	0.63	0.67	0.65	18
Group Setting Environment	0.89	0.89	0.89	28
Reverberant Spaces Environment	0.79	0.69	0.73	16
Telephone Conversations Environment	0.81	0.94	0.87	18
Accuracy	-	-	0.95	577
Macro Average	0.87	0.85	0.85	577
Weighted Average	0.95	0.95	0.95	577

Table 3 presents the performance metrics for the environmental sound classification across various types of listening conditions. Each row corresponds to a specific environment type, while the columns display precision, recall, F1-score, and support (the number of samples) for each category.

Precision indicates how many of the predicted positive results were actually correct, out of all the positive results the model predicted. This shows how well the model can spot instances of a specific type of environment without mistakenly labeling other things. Recall, on the other hand, tells us how many of the actual positive instances the model managed to find out of all the positive instances there actually were. This shows the sensitivity of the model for picking up instances of a specific environment type.

$$\text{Precision} = \frac{\text{True Positive}}{\text{True Positive}+\text{False Positive}} = \frac{\text{TP}}{\text{TP}+\text{FP}} \qquad 2$$

$$\text{Recall} = \frac{\text{True Positive}}{\text{True Positive }+\text{False Negative}} = \frac{\text{TP}}{\text{TP}+\text{FN}} \qquad 3$$

The F1-score is like a blend of precision and recall, finding a balance between them. This is is especially helpful when some types of things are more common than others in the data. Support indicates the number of times each type of thing actually appeared in the dataset. This helps us better understand precision, recall, and F1-score by providing context.

$$\text{F1-score} = \frac{2(\text{Precision} * \text{Recall})}{\text{Precision} + \text{Recall}} = \frac{\text{TP}}{\text{TP} + 1/2(\text{FP} + \text{FN})} \qquad 4$$

For instance, in the "Quiet Environment" category, we observed perfect precision, recall, and an F1-score of 1.00, indicating that the model correctly classified all instances of quiet environments without any false positives or negatives [24]. However, in environments like "Restaurant Noise" and "Street Noise," where there may be more variability and overlap in audio characteristics, we see slightly lower scores, particularly in recall, indicating that the model may miss some instances of these environments.

Furthermore, the table includes the overall accuracy as well as the macro-average and weighted-average scores. Accuracy shows how correct the model is overall, considering all categories. The macro-average and weighted-average metrics provide a summarized view of precision, recall, and F1-score for all categories, considering either equal importance to each category or taking into account how common each category is. These summarized metrics provide a thorough evaluation of the performance of the model on the entire dataset, considering both specific categories and overall performance.

$$\text{Accuracy} = \frac{\text{Number of correct predictions}}{\text{Total number of predictions}} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{TN} + \text{FP} + \text{FN}} \qquad 5$$

Where TP signifies True Positives, TN denotes True Negatives, FP represents False Positives, and FN stands for False Negatives.

Table 4: RNN Model Training and Testing Accuracy

RNN Model	**Accuracy**
Training	98.50%
Testing	94.97%

Table 4 shows the performance metrics of an (RNN) model, detailing its accuracy during both the training and testing phases. During the training process, the RNN model achieved an impressive accuracy of 98.50%, indicating its proficiency in correctly classifying audio samples within the dataset on which it was trained. This high accuracy rate underscores the ability of the model to effectively learn and capture patterns present in the training data, enabling it to make accurate predictions during the training phase.

Moving on to the testing phase, where the model's performance was evaluated on unseen data, an accuracy of 94.97% was observed. Although marginally lower than the training accuracy, this testing accuracy remains notably high, signifying the robustness and generalization capability of the model. Despite being exposed to new and unseen audio samples, the RNN model continued to exhibit strong predictive power, accurately classifying the majority of the test data. This indicates that the model has effectively learned relevant features and patterns from the training data and is capable of applying this knowledge to new instances, making it a reliable tool for detecting and classifying diverse listening conditions, which is particularly beneficial for individuals with hearing impairments.

In the confusion matrix as shown in Fig 4, every row shows the real categories, and each column shows the estimated model. The numbers in the cells of the matrix indicate the number of times the model put something in each combination of real and guessed categories. This helps us to carefully examine how well the model is doing, such as how often it gets each category right and where it makes mistakes.

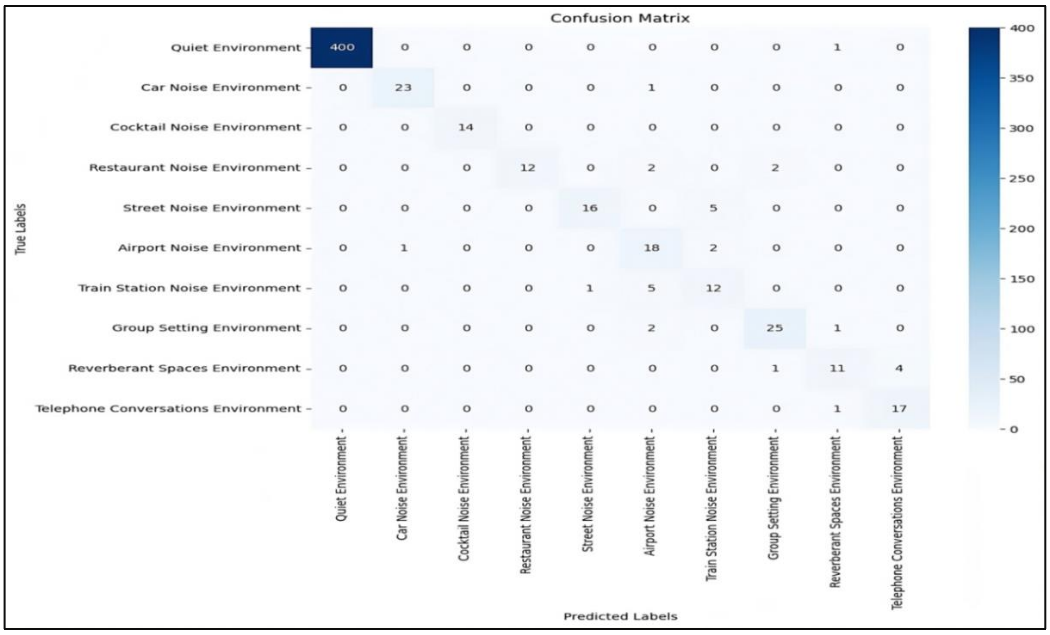


Fig.4: Confusion matrix

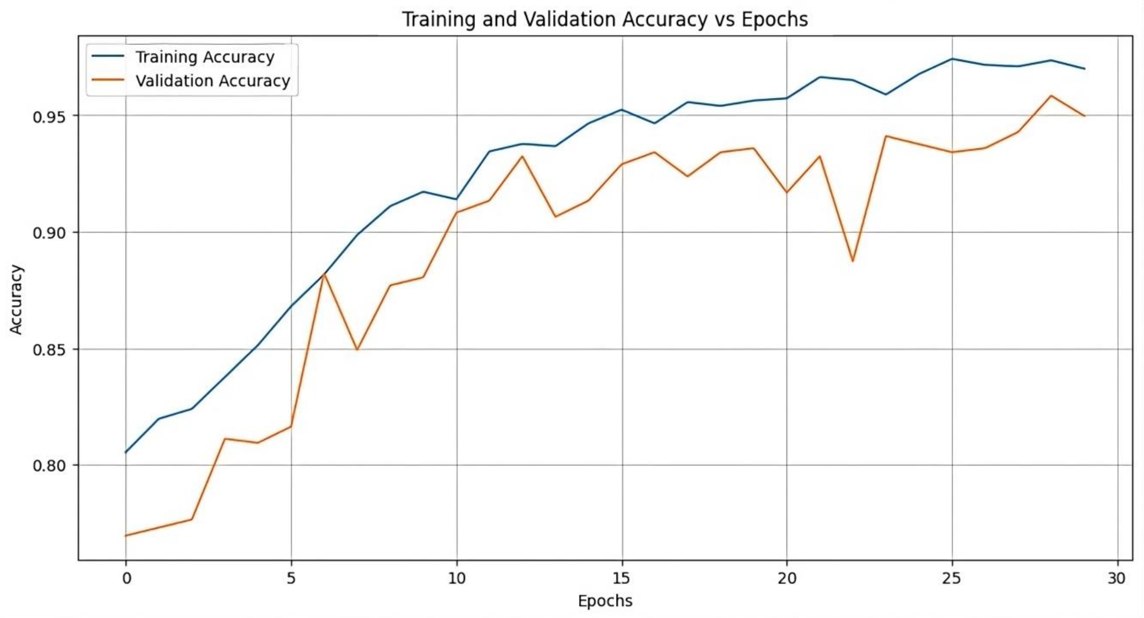


Fig.5: Training and Validation Accuracy vs Epochs

Fig. 5 displays the relationship between the training and validation accuracies over the course of the training epochs. This illustrates how the accuracy of the model changes as it undergoes training, with separate lines or curves representing the accuracy of the training and validation datasets. This visualization allows for the assessment of the model performance and generalization ability throughout the training process.

The training accuracy of 98.50% and testing accuracy of 94.97% represent the performance of the Recurrent Neural Network (RNN) model after training it for 30 epochs. During the training process, the model passes through the training data multiple times, which are called epochs. In this case, it underwent 30 epochs. Each time it goes through the data, the model tweaks its internal settings (such as weights and biases) based on what it has learned, trying to get closer to the correct answers. The training accuracy, which is 98.50%, shows how often the model obtained the correct answers when predicting the labels of the training data after all epochs. This high accuracy indicates that the model has learned well and remembered patterns from the training data.

The real measure of how well the model works comes when testing it with new data that has not been seen before. This occurs during the testing phase. The testing accuracy, which is 94.97%, shows how often the model obtains the correct answers when it is asked questions about this new data, which is different from what it is trained on. This means that the model does es not simply repeat what it learned from the training data; it also determines how to give good answers to new questions.

It is common for the testing accuracy to be slightly lower than the training one. This typically occurs because of overfitting or noise in the training data. Overfitting occurs when a model is too good to pick up tiny details or random patterns in the training data that do not really matter. This can cause it to not work as well when it asking questions about new data. However, the relatively small difference between the training and testing accuracies (3.53%) suggests that the model is well-generalized and effectively discriminates between different classes, even when presented with new examples. Overall, the model demonstrated a strong performance after training for 30 epochs, achieving high accuracy on both the training and testing datasets.

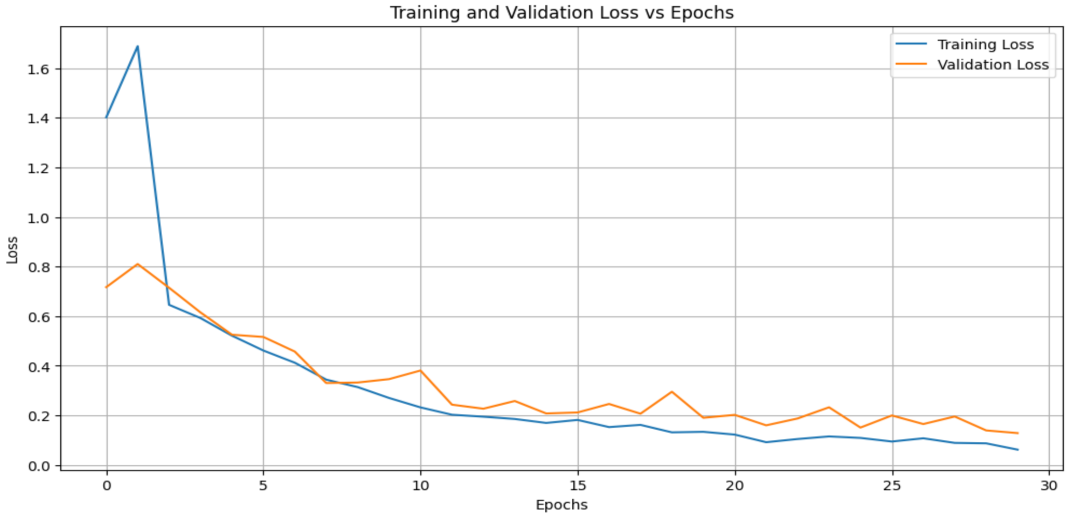
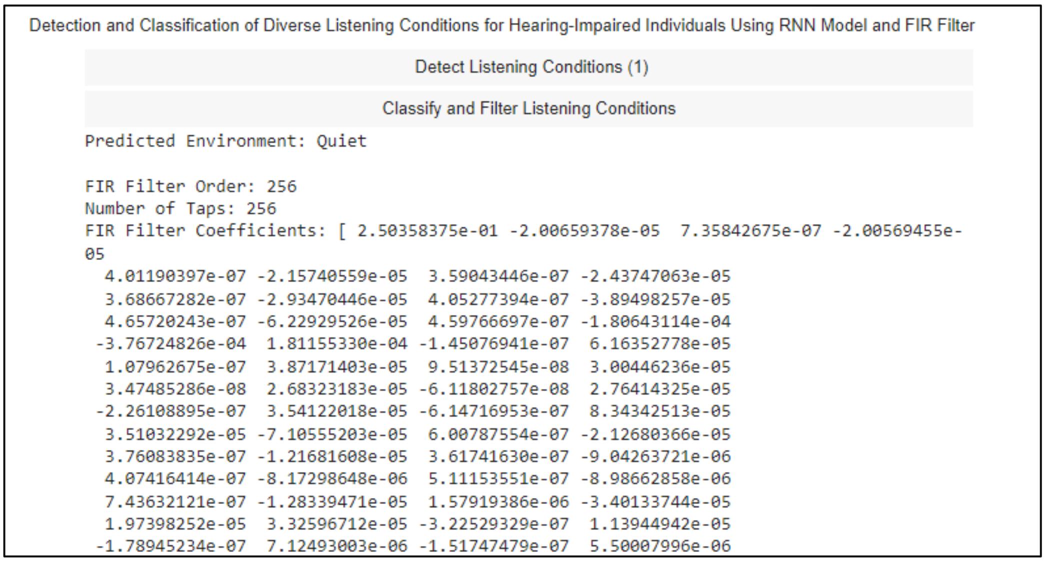


Fig.6: Training and Validation Loss vs Epochs

Fig. 6 depicts the variations in training and validation losses at several epochs during the training process. This visualization shows how the loss, which reflects the gap between the predicted and actual values, changes as the model learns from the training data. The comparison of training and validation loss assesses the model's capacity to generalize to new, previously unknown data.

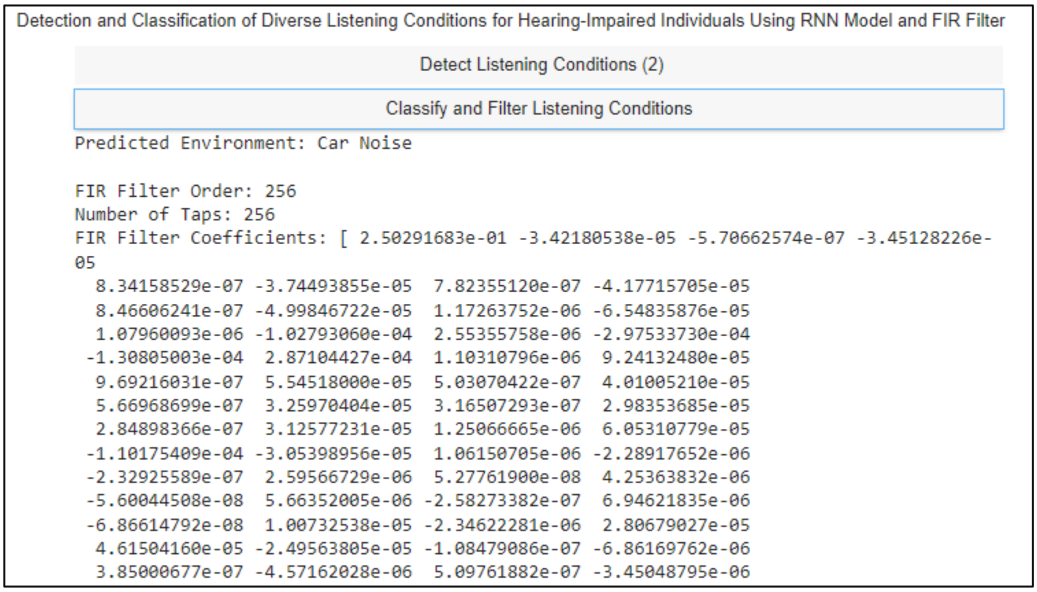
When evaluating the efficacy and learning curve of machine learning models, like recurrent neural networks (RNNs), over a 30-epoch period, training and validation loss metrics provide critical information. As the difference between the true and predicted labels in the training dataset is represented by the training loss, the model's goal is to minimize it. Decreasing the training loss indicates effective learning of the underlying patterns. Similarly, the validation loss, computed on unseen data, should ideally decrease over epochs, reflecting the model's ability to generalize its learned patterns. Consistent decreases in both training and validation losses signify effective learning and generalization, ensuring the accuracy of the model on new data.


Detection and Classification of Diverse Listening Conditions for Hearing-Impaired Individuals Using RNN Model and FIR Filter
Detect Listening Conditions (1)
Classify and Filter Listening Conditions

```
Predicted Environment: Quiet

FIR Filter Order: 256
Number of Taps: 256
FIR Filter Coefficients: [ 2.50358375e-01 -2.00659378e-05  7.35842675e-07 -2.00569455e-
05
  4.01190397e-07 -2.15740559e-05  3.59043446e-07 -2.43747063e-05
  3.68667282e-07 -2.93470446e-05  4.05277394e-07 -3.89498257e-05
  4.65720243e-07 -6.22929526e-05  4.59766697e-07 -1.80643114e-04
 -3.76724826e-04  1.81155330e-04 -1.45076941e-07  6.16352778e-05
  1.07962675e-07  3.87171403e-05  9.51372545e-08  3.00446236e-05
  3.47485286e-08  2.68323183e-05 -6.11802757e-08  2.76414325e-05
 -2.26108895e-07  3.54122018e-05 -6.14716953e-07  8.34342513e-05
  3.51032292e-05 -7.10555203e-05  6.00787554e-07 -2.12680366e-05
  3.76083835e-07 -1.21681608e-05  3.61741630e-07 -9.04263721e-06
  4.07416414e-07 -8.17298648e-06  5.11153551e-07 -8.98662858e-06
  7.43632121e-07 -1.28339471e-05  1.57919386e-06 -3.40133744e-05
  1.97398252e-05  3.32596712e-05 -3.22529329e-07  1.13944942e-05
 -1.78945234e-07  7.12493003e-06 -1.51747479e-07  5.50007996e-06
```

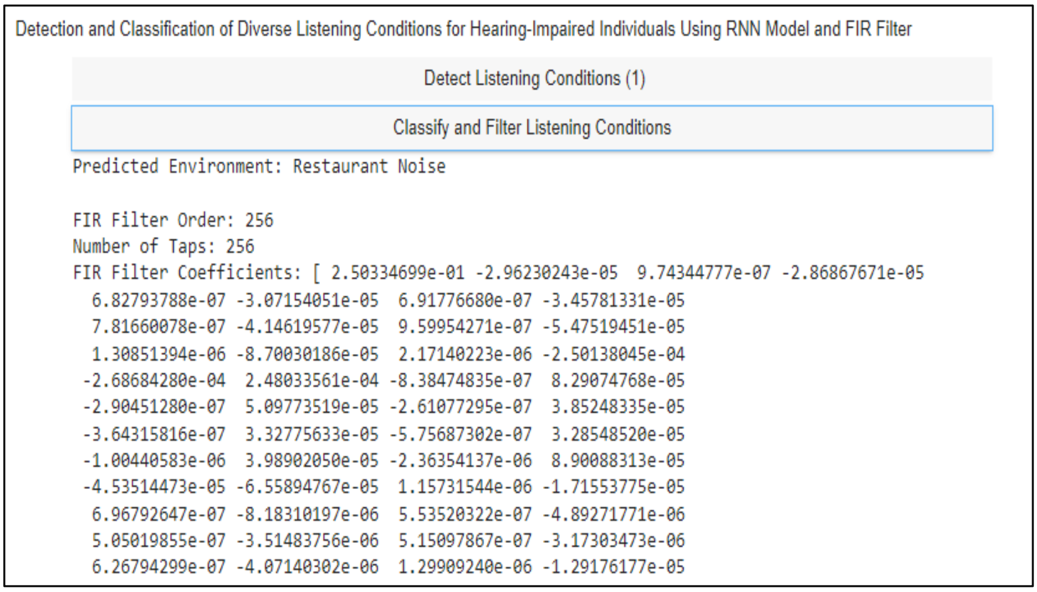
Fig.7: Predicted Environment is Quiet Environment


Detection and Classification of Diverse Listening Conditions for Hearing-Impaired Individuals Using RNN Model and FIR Filter
Detect Listening Conditions (2)
Classify and Filter Listening Conditions

```
Predicted Environment: Car Noise

FIR Filter Order: 256
Number of Taps: 256
FIR Filter Coefficients: [ 2.50291683e-01 -3.42180538e-05 -5.70662574e-07 -3.45128226e-
05
  8.34158529e-07 -3.74493855e-05  7.82355120e-07 -4.17715705e-05
  8.46606241e-07 -4.99846722e-05  1.17263752e-06 -6.54835876e-05
  1.07960093e-06 -1.02793060e-04  2.55355758e-06 -2.97533730e-04
 -1.30805003e-04  2.87104427e-04  1.10310796e-06  9.24132480e-05
  9.69216031e-07  5.54518000e-05  5.03070422e-07  4.01005210e-05
  5.66968699e-07  3.25970404e-05  3.16507293e-07  2.98353685e-05
  2.84898366e-07  3.12577231e-05  1.25066665e-06  6.05310779e-05
 -1.10175409e-04 -3.05398956e-05  1.06150705e-06 -2.28917652e-06
 -2.32925589e-07  2.59566729e-06  5.27761900e-08  4.25363832e-06
 -5.60044508e-08  5.66352005e-06 -2.58273382e-07  6.94621835e-06
 -6.86614792e-08  1.00732538e-05 -2.34622281e-06  2.80679027e-05
  4.61504160e-05 -2.49563805e-05 -1.08479086e-07 -6.86169762e-06
  3.85000677e-07 -4.57162028e-06  5.09761882e-07 -3.45048795e-06
```

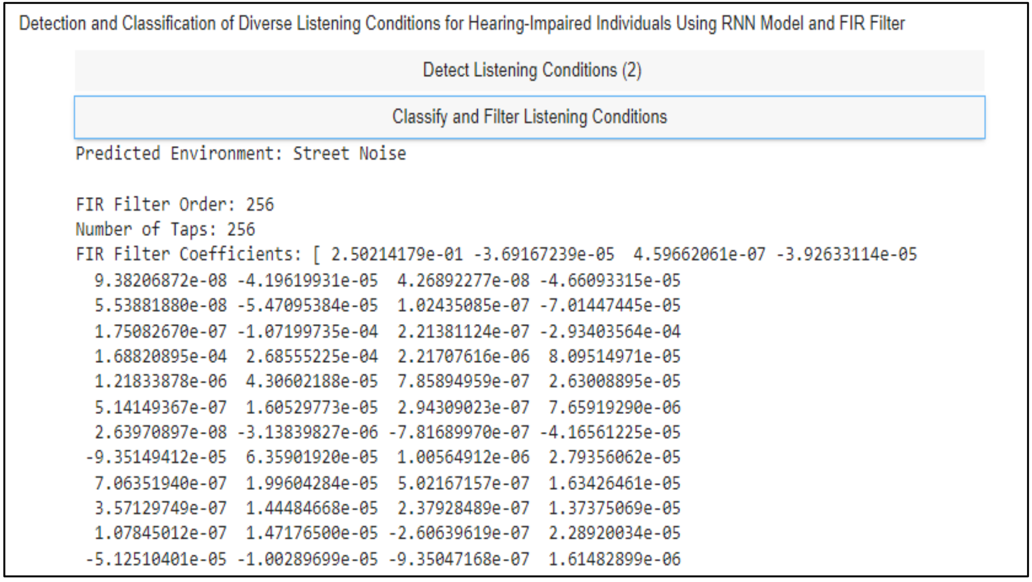
Fig.8: Predicted Environment is Car Noise Environment


Detection and Classification of Diverse Listening Conditions for Hearing-Impaired Individuals Using RNN Model and FIR Filter
Detect Listening Conditions (1)
Classify and Filter Listening Conditions

```
Predicted Environment: Restaurant Noise

FIR Filter Order: 256
Number of Taps: 256
FIR Filter Coefficients: [ 2.50334699e-01 -2.96230243e-05  9.74344777e-07 -2.86867671e-05
  6.82793788e-07 -3.07154051e-05  6.91776680e-07 -3.45781331e-05
  7.81660078e-07 -4.14619577e-05  9.59954271e-07 -5.47519451e-05
  1.30851394e-06 -8.70030186e-05  2.17140223e-06 -2.50138045e-04
 -2.68684280e-04  2.48033561e-04 -8.38474835e-07  8.29074768e-05
 -2.90451280e-07  5.09773519e-05 -2.61077295e-07  3.85248335e-05
 -3.64315816e-07  3.32775633e-05 -5.75687302e-07  3.28548520e-05
 -1.00440583e-06  3.98902050e-05 -2.36354137e-06  8.90088313e-05
 -4.53514473e-05 -6.55894767e-05  1.15731544e-06 -1.71553775e-05
  6.96792647e-07 -8.18310197e-06  5.53520322e-07 -4.89271771e-06
  5.05019855e-07 -3.51483756e-06  5.15097867e-07 -3.17303473e-06
  6.26794299e-07 -4.07140302e-06  1.29909240e-06 -1.29176177e-05
```

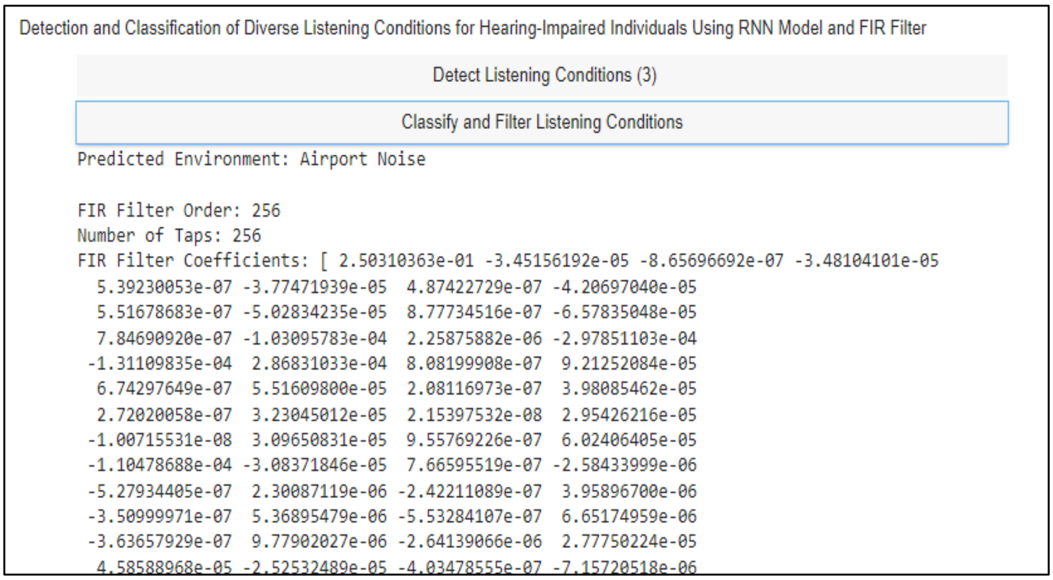
Fig.9: Predicted Environment is Restaurant Noise Environment


Detection and Classification of Diverse Listening Conditions for Hearing-Impaired Individuals Using RNN Model and FIR Filter
Detect Listening Conditions (2)
Classify and Filter Listening Conditions

```
Predicted Environment: Street Noise

FIR Filter Order: 256
Number of Taps: 256
FIR Filter Coefficients: [ 2.50214179e-01 -3.69167239e-05  4.59662061e-07 -3.92633114e-05
  9.38206872e-08 -4.19619931e-05  4.26892277e-08 -4.66093315e-05
  5.53881880e-08 -5.47095384e-05  1.02435085e-07 -7.01447445e-05
  1.75082670e-07 -1.07199735e-04  2.21381124e-07 -2.93403564e-04
  1.68820895e-04  2.68555225e-04  2.21707616e-06  8.09514971e-05
  1.21833878e-06  4.30602188e-05  7.85894959e-07  2.63008895e-05
  5.14149367e-07  1.60529773e-05  2.94309023e-07  7.65919290e-06
  2.63970897e-08 -3.13839827e-06 -7.81689970e-07 -4.16561225e-05
 -9.35149412e-05  6.35901920e-05  1.00564912e-06  2.79356062e-05
  7.06351940e-07  1.99604284e-05  5.02167157e-07  1.63426461e-05
  3.57129749e-07  1.44484668e-05  2.37928489e-07  1.37375069e-05
  1.07845012e-07  1.47176500e-05 -2.60639619e-07  2.28920034e-05
 -5.12510401e-05 -1.00289699e-05 -9.35047168e-07  1.61482899e-06
```

Fig.10: Predicted Environment is Street Noise Environment

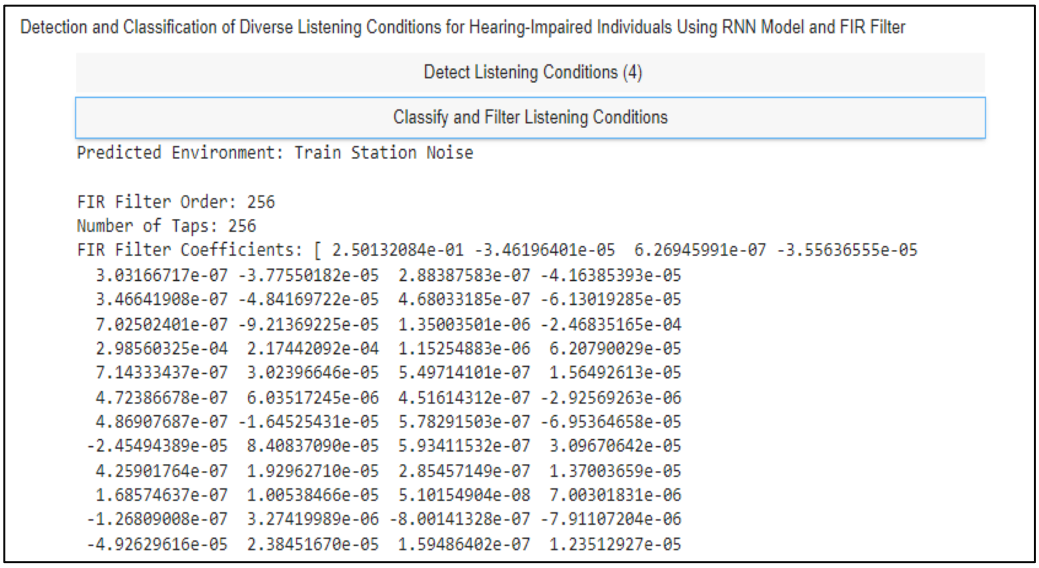


Detection and Classification of Diverse Listening Conditions for Hearing-Impaired Individuals Using RNN Model and FIR Filter
Detect Listening Conditions (3)
Classify and Filter Listening Conditions

```
Predicted Environment: Airport Noise

FIR Filter Order: 256
Number of Taps: 256
FIR Filter Coefficients: [ 2.50310363e-01 -3.45156192e-05 -8.65696692e-07 -3.48104101e-05
  5.39230053e-07 -3.77471939e-05  4.87422729e-07 -4.20697040e-05
  5.51678683e-07 -5.02834235e-05  8.77734516e-07 -6.57835048e-05
  7.84690920e-07 -1.03095783e-04  2.25875882e-06 -2.97851103e-04
 -1.31109835e-04  2.86831033e-04  8.08199908e-07  9.21252084e-05
  6.74297649e-07  5.51609800e-05  2.08116973e-07  3.98085462e-05
  2.72020058e-07  3.23045012e-05  2.15397532e-08  2.95426216e-05
 -1.00715531e-08  3.09650831e-05  9.55769226e-07  6.02406405e-05
 -1.10478688e-04 -3.08371846e-05  7.66595519e-07 -2.58433999e-06
 -5.27934405e-07  2.30087119e-06 -2.42211089e-07  3.95896700e-06
 -3.50999971e-07  5.36895479e-06 -5.53284107e-07  6.65174959e-06
 -3.63657929e-07  9.77902027e-06 -2.64139066e-06  2.77750224e-05
  4.58588968e-05 -2.52532489e-05 -4.03478555e-07 -7.15720518e-06
```

Fig.11: Predicted Environment is Airport Noise Environment

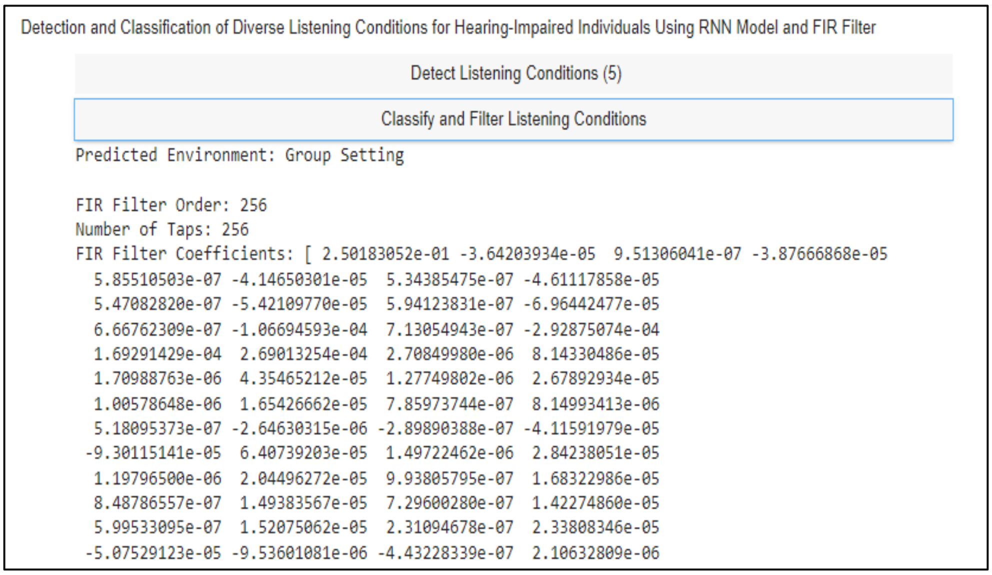


Detection and Classification of Diverse Listening Conditions for Hearing-Impaired Individuals Using RNN Model and FIR Filter
Detect Listening Conditions (4)
Classify and Filter Listening Conditions

```
Predicted Environment: Train Station Noise

FIR Filter Order: 256
Number of Taps: 256
FIR Filter Coefficients: [ 2.50132084e-01 -3.46196401e-05  6.26945991e-07 -3.55636555e-05
  3.03166717e-07 -3.77550182e-05  2.88387583e-07 -4.16385393e-05
  3.46641908e-07 -4.84169722e-05  4.68033185e-07 -6.13019285e-05
  7.02502401e-07 -9.21369225e-05  1.35003501e-06 -2.46835165e-04
  2.98560325e-04  2.17442092e-04  1.15254883e-06  6.20790029e-05
  7.14333437e-07  3.02396646e-05  5.49714101e-07  1.56492613e-05
  4.72386678e-07  6.03517245e-06  4.51614312e-07 -2.92569263e-06
  4.86907687e-07 -1.64525431e-05  5.78291503e-07 -6.95364658e-05
 -2.45494389e-05  8.40837090e-05  5.93411532e-07  3.09670642e-05
  4.25901764e-07  1.92962710e-05  2.85457149e-07  1.37003659e-05
  1.68574637e-07  1.00538466e-05  5.10154904e-08  7.00301831e-06
 -1.26809008e-07  3.27419989e-06 -8.00141328e-07 -7.91107204e-06
 -4.92629616e-05  2.38451670e-05  1.59486402e-07  1.23512927e-05
```

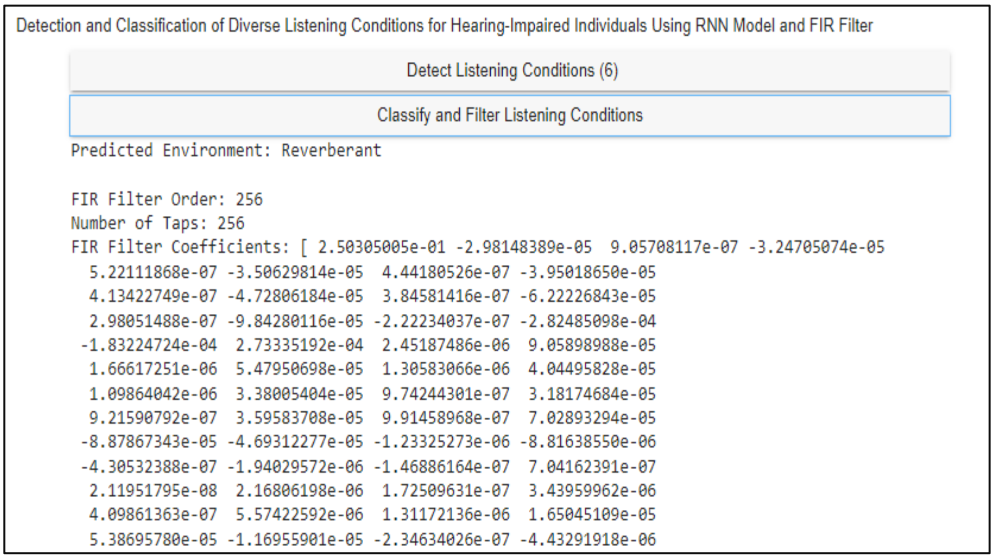
Fig.12: shows the Predicted Environment is Train Station Noise Environment


Detection and Classification of Diverse Listening Conditions for Hearing-Impaired Individuals Using RNN Model and FIR Filter
Detect Listening Conditions (5)
Classify and Filter Listening Conditions

```
Predicted Environment: Group Setting

FIR Filter Order: 256
Number of Taps: 256
FIR Filter Coefficients: [ 2.50183052e-01 -3.64203934e-05  9.51306041e-07 -3.87666868e-05
  5.85510503e-07 -4.14650301e-05  5.34385475e-07 -4.61117858e-05
  5.47082820e-07 -5.42109770e-05  5.94123831e-07 -6.96442477e-05
  6.66762309e-07 -1.06694593e-04  7.13054943e-07 -2.92875074e-04
  1.69291429e-04  2.69013254e-04  2.70849980e-06  8.14330486e-05
  1.70988763e-06  4.35465212e-05  1.27749802e-06  2.67892934e-05
  1.00578648e-06  1.65426662e-05  7.85973744e-07  8.14993413e-06
  5.18095373e-07 -2.64630315e-06 -2.89890388e-07 -4.11591979e-05
 -9.30115141e-05  6.40739203e-05  1.49722462e-06  2.84238051e-05
  1.19796500e-06  2.04496272e-05  9.93805795e-07  1.68322986e-05
  8.48786557e-07  1.49383567e-05  7.29600280e-07  1.42274860e-05
  5.99533095e-07  1.52075062e-05  2.31094678e-07  2.33808346e-05
 -5.07529123e-05 -9.53601081e-06 -4.43228339e-07  2.10632809e-06
```

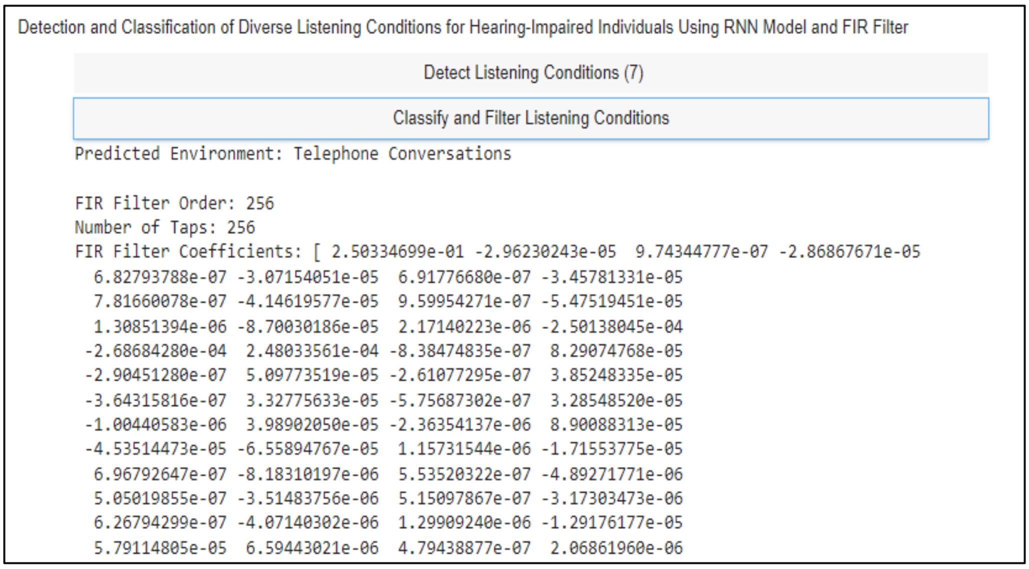
Fig.13: Predicted Environment is Group Setting Environment


Detection and Classification of Diverse Listening Conditions for Hearing-Impaired Individuals Using RNN Model and FIR Filter
Detect Listening Conditions (6)
Classify and Filter Listening Conditions

```
Predicted Environment: Reverberant

FIR Filter Order: 256
Number of Taps: 256
FIR Filter Coefficients: [ 2.50305005e-01 -2.98148389e-05  9.05708117e-07 -3.24705074e-05
  5.22111868e-07 -3.50629814e-05  4.44180526e-07 -3.95018650e-05
  4.13422749e-07 -4.72806184e-05  3.84581416e-07 -6.22226843e-05
  2.98051488e-07 -9.84280116e-05 -2.22234037e-07 -2.82485098e-04
 -1.83224724e-04  2.73335192e-04  2.45187486e-06  9.05898988e-05
  1.66617251e-06  5.47950698e-05  1.30583066e-06  4.04495828e-05
  1.09864042e-06  3.38005404e-05  9.74244301e-07  3.18174684e-05
  9.21590792e-07  3.59583708e-05  9.91458968e-07  7.02893294e-05
 -8.87867343e-05 -4.69312277e-05 -1.23325273e-06 -8.81638550e-06
 -4.30532388e-07 -1.94029572e-06 -1.46886164e-07  7.04162391e-07
  2.11951795e-08  2.16806198e-06  1.72509631e-07  3.43959962e-06
  4.09861363e-07  5.57422592e-06  1.31172136e-06  1.65045109e-05
  5.38695780e-05 -1.16955901e-05 -2.34634026e-07 -4.43291918e-06
```

Fig.14: Predicted Environment is Reverberant Environment.



Detection and Classification of Diverse Listening Conditions for Hearing-Impaired Individuals Using RNN Model and FIR Filter

Detect Listening Conditions (7)
Classify and Filter Listening Conditions

```
Predicted Environment: Telephone Conversations

FIR Filter Order: 256
Number of Taps: 256
FIR Filter Coefficients: [ 2.50334699e-01 -2.96230243e-05  9.74344777e-07 -2.86867671e-05
  6.82793788e-07 -3.07154051e-05  6.91776680e-07 -3.45781331e-05
  7.81660078e-07 -4.14619577e-05  9.59954271e-07 -5.47519451e-05
  1.30851394e-06 -8.70030186e-05  2.17140223e-06 -2.50138045e-04
 -2.68684280e-04  2.48033561e-04 -8.38474835e-07  8.29074768e-05
 -2.90451280e-07  5.09773519e-05 -2.61077295e-07  3.85248335e-05
 -3.64315816e-07  3.32775633e-05 -5.75687302e-07  3.28548520e-05
 -1.00440583e-06  3.98902050e-05 -2.36354137e-06  8.90088313e-05
 -4.53514473e-05 -6.55894767e-05  1.15731544e-06 -1.71553775e-05
  6.96792647e-07 -8.18310197e-06  5.53520322e-07 -4.89271771e-06
  5.05019855e-07 -3.51483756e-06  5.15097867e-07 -3.17303473e-06
  6.26794299e-07 -4.07140302e-06  1.29909240e-06 -1.29176177e-05
  5.79114805e-05  6.59443021e-06  4.79438877e-07  2.06861960e-06
```

Fig.15: Predicted Environment is Telephone Conversations Environment

Fig. 7-15 reveals that the Recurrent Neural Network (RNN) model identified the environment as “Quiet Environment, Car Noise Environment, Restaurant Noise Environment, Street Noise Environment, Airport Noise Environment, Train Station Noise Environment, Group Setting Environment, Reverberant Environment, and Telephone Conversation Environment, respectively. This decision relied on a Finite Impulse Response (FIR) filter with 256 taps and a FIR filter order of 256. By meticulously examining the audio signals with the assistance of the FIR filter, the RNN model accurately labelled the environment with the respective environment type. This demonstrates how combining advanced signal processing techniques with deep learning methods can effectively classify environments.

5. Conclusion and Future Scope

This study proposes an innovative approach that combines recurrent neural network (RNN) models with finite impulse response (FIR) filtering to effectively detect and classify diverse listening conditions experienced by individuals with hearing impairment. By leveraging machine learning and signal processing techniques, this methodology provides a comprehensive solution to address the unique auditory challenges encountered by this population. Through the integration of RNN models, which are capable of capturing temporal dependencies in sequential data, and dynamic FIR filtering, which enhances audio quality based on predicted environments, this model accurately categorizes diverse listening conditions with a training accuracy of 98.50% and testing accuracy of 94.97%. This model empowers individuals with hearing impairment by offering personalized solutions that enhance their auditory experiences and improve their overall quality of life, fostering inclusivity and accessibility in auditory environments.

Future research opportunities for this method include investigating advanced model architectures such as attention-based RNNs or transformer models to improve classification performance and robustness. Developing real-time processing capabilities and customizable adaptation approaches could allow for more tailored responses to individual user needs.

Integrating multimodal sensory signals and conducting long-term evaluation studies can improve the user experience and provide information about the system's long-term effectiveness in real-world scenarios. This continuing work seeks to enhance assistive technology for people with hearing loss, ultimately improving accessibility and quality of life for this population.

Abbreviations

ML: Machine Learning
MFCCs: Mel-Frequency Cepstral Coefficients
RNN: Recurrent Neural Network
FIR: Finite Impulse Response
ESC: Environmental Sound Classification
FFT: Fast Fourier Transform
LSTM: Long Short-Term Memory

Declarations:

Availability of data and materials

The datasets used in this study were sourced from the NOIZEUS database and the AURORA-2 corpus dataset. A noisy speech corpus (NOIZEUS) was developed to facilitate comparison of speech enhancement algorithms among research groups. The noisy database contains 30 IEEE sentences (produced by three male and three female speakers) corrupted by eight different real-world noises at different SNRs. The noise was taken from the AURORA database and includes suburban train noise, babble, car, exhibition hall, restaurant, street, airport and train-station noise. This dataset is available on the following URL.

https://ecs.utdallas.edu/loizou/speech/noizeus/
https://www.kaggle.com/datasets.

Conflicts of Interest

The authors declare that they have no Conflicts of Interest.

Funding

This research did not receive any specific grant from funding agencies in the public, commercial, or not-for-profit sectors.

Authors' contributions

Sunilkumar M. Hattaraki, collected the data, analyzed the data, implemented the proposed work and drafted the complete manuscript. Shankarayya G. Kambalimath defined the problem statement and provided critical reviews. All authors read and approved the final manuscript.

Acknowledgements

I, Sunilkumar M. Hattaraki, would like to express my gratitude to the Research Centre, Department of Electronics and Communication Engineering at Basaveshwar Engineering College, Bagalkote, Karnataka, India.

References

[1]. Zaman, Khalid, Melike Sah, Cem Direkoglu, and Masashi Unoki. "A Survey of Audio Classification Using Deep Learning." IEEE Access (2023).

[2]. Shao, Xi, Changsheng Xu, and Mohan S. Kankanhalli. "Applying neural network on the content-based audio classification." Fourth International Conference on Information, Communications and Signal Processing, 2003 and the Fourth Pacific Rim Conference on Multimedia. Proceedings of the 2003 Joint. Vol. 3. IEEE, 2003.

[3]. Freeman, C., R. D. Dony, and S. M. Areibi. "Audio environment classification for hearing aids using artificial neural networks with windowed input." 2007 IEEE Symposium on Computational Intelligence in Image and Signal Processing. IEEE, 2007.

[4]. Mitra, Vikramjit, and Chia-Jiu Wang. "Content based audio classification: a neural network approach." Soft Computing 12 (2008): 639-646.

[5]. Veena, S., Aravindhar, D.J. Sound Classification System Using Deep Neural Networks for Hearing Impaired People. Wireless Pers Commun 126, 385–399 (2022). https://doi.org/10.1007/s11277-022-09750-7.

[6]. Khamparia, A., Gupta, D., Nguyen, N.G., Khanna, A., Pandey, B. and Tiwari, P., 2019. Sound classification using convolutional neural network and tensor deep stacking network. IEEE Access, 7, pp.7717-7727.

[7]. Su, Yu, Ke Zhang, Jingyu Wang, and Kurosh Madani. 2019. "Environment Sound Classification Using a Two-Stream CNN Based on Decision-Level Fusion" Sensors 19, no. 7: 1733. https://doi.org/10.3390/s19071733.

[8]. Yan Zhao, DeLiang Wang, Eric M. Johnson, Eric W. Healy; A deep learning based segregation algorithm to increase speech intelligibility for hearing-impaired listeners in reverberant-noisy conditions. J. Acoust. Soc. Am. 1 September 2018; 144 (3): 1627–1637.

[9]. Mushtaq, Zohaib, and Shun-Feng Su. "Environmental sound classification using a regularized deep convolutional neural network with data augmentation." Applied Acoustics 167 (2020): 107389.

[10]. Mushtaq, Zohaib, Shun-Feng Su, and Quoc-Viet Tran. "Spectral images based environmental sound classification using CNN with meaningful data augmentation." Applied Acoustics 172 (2021): 107581.

[11]. Toffa, Ohini Kafui, and Max Mignotte. "Environmental Sound Classification Using Local Binary Pattern and Audio Features Collaboration." IEEE TRANSACTIONS ON MULTIMEDIA 23 (2021).

[12]. Ali, Yaseen Hadi, Rozeha A. Rashid, and Siti Zaleha Abdul Hamid. "A machine learning for environmental noise classification in smart cities." Indonesian Journal of Electrical Engineering and Computer Science 25.3 (2022): 1777-1786.

[13]. Demir, Fatih, Daban Abdulsalam Abdullah, and Abdulkadir Sengur. "A new deep CNN model for environmental sound classification." IEEE Access 8 (2020): 66529-66537.

[14]. Z. Chi, Y. Li and C. Chen, "Deep Convolutional Neural Network Combined with Concatenated Spectrogram for Environmental Sound Classification," 2019 IEEE 7th International Conference on Computer Science and Network Technology (ICCSNT), Dalian, China, 2019, pp. 251-254, doi: 10.1109/ICCSNT47585.2019.8962462.

[15]. Hu, Yi, and Philipos C. Loizou. "Evaluation of objective quality measures for speech enhancement." IEEE Transactions on audio, speech, and language processing 16.1 (2007): 229-238.

[16]. Ma, Jianfen, Yi Hu, and Philipos C. Loizou. "Objective measures for predicting speech intelligibility in noisy conditions based on new band-importance functions." The Journal of the Acoustical Society of America 125.5 (2009): 3387-3405.

[17]. https://www.kaggle.com/datasets/chrisfilo/urbansound8k.

[18]. Siddharth, S., C. Rajakumaran, and A. Revathi. "Impact of filtering on the performance of bird classification system." International Journal of Pure and Applied Mathematics 119.12 (2018): 13609-13615.

[19]. Park, Gyuseok, et al. "Speech enhancement for hearing aids with deep learning on environmental noises." Applied Sciences 10.17 (2020): 6077.

[20]. Sharma, Jivitesh, Ole-Christoffer Granmo, and Morten Goodwin. "Environment sound classification using multiple feature channels and attention based deep convolutional neural network." (2020).

[21]. da Silva, Bruno, et al. "Evaluation of classical machine learning techniques towards urban sound recognition on embedded systems." Applied Sciences 9.18 (2019): 3885.

[22]. A. Graves, A. -r. Mohamed and G. Hinton, "Speech recognition with deep recurrent neural networks," 2013 IEEE International Conference on Acoustics, Speech and Signal Processing, Vancouver, BC, Canada, 2013, pp. 6645-6649, doi: 10.1109/ICASSP.2013.6638947.

[23]. Alec Zhang. "A Novel Eye-tracking and Audio Hybrid System for Autism Spectrum Disorder Early Detection" , 2023 IEEE 3rd International Conference on Data Science and Computer Application (ICDSCA), 2023.

[24]. "Artificial Intelligence: Theory and Applications" , Springer Science and Business Media LLC, 2024.

[25]. Herrera-Ortiz, A. D., et al. "An Entropy-Based Computational Classifier for Positive and Negative Emotions in Voice Signals." International Congress of Telematics and Computing. Cham: Springer International Publishing, 2022.

[26]. Zhao, Yan, et al. "A deep learning based segregation algorithm to increase speech intelligibility for hearing-impaired listeners in reverberant-noisy conditions." The Journal of the Acoustical Society of America 144.3 (2018): 1627-1637.

[27]. Graves, Alex, Abdel-rahman Mohamed, and Geoffrey Hinton. "Speech recognition with deep recurrent neural networks." 2013 IEEE international conference on acoustics, speech and signal processing. IEEE, 2013.