Dr. M. S. Sheshgiri Campus, Belagavi

Department of
Electronics and Communication Engineering

Mini Project Report

on

ENHANCED AUDIO NOISE REDUCTION SYSTEM

By:

1. **Rohan Anvekar** USN: 02FE21BEC074
2. **Rohit M Hosalli** USN: 02FE21BEC075
3. **Samruddhi** USN: 02FE21BEC083
4. **Ujwala V T** USN: 02FE21BEC105

Semester: V, 2023-2024

Under the Guidance of

Dr. Prabhakar Manage

KLE Technological University,
Dr. M. S. Sheshgiri College of Engineering and Technology
BELAGAVI-590 008

2023-2024

Dr. M. S. Sheshgiri Campus, Belagavi

DEPARTMENT OF ELECTRONICS AND COMMUNICATION ENGINEERING

CERTIFICATE

This is to certify that the project entitled **"ENHANCED AUDIO NOISE REDUCTION SYSTEM"** is a bonafide work carried out by the student team of **"Rohan Anvekar 02FE21BEC074, Rohit M Hosalli 02FE21BEC075, Samruddhi 02FE21BEC083, Ujwala V T 02FE21BEC105"**. The project report has been approved as it satisfies the requirements with respect to the mini project work prescribed by the university curriculum for B.E. (V Semester) in the Department of Electronics and Communication Engineering of KLE Technological University Dr. M. S. Sheshgiri CET Belagavi campus for the academic year 2023-2024.

Dr. Prabhakar Manage **Guide**	**Dr. Dattaprasad A. Torse** **Head of Department**	**Dr. S. F. Patil** **Principal**

External Viva:

Name of Examiners	**Signature with date**
1.	
2.	

ACKNOWLEDGMENT

I want to thank Prof. Prabhakar Manage for helping me a lot with our project on "Enhanced Audio Noise Reduction System." He gave me great advice, supported us all the way, and worked really hard. His teaching and guidance made our project better. Prof. Manage was not just a teacher but also a mentor, making our learning more practical. He was patient and encouraging when things got tough, and his help was not just about the technical stuff but also about working well together as a team. I'm really thankful for his support, and our project wouldn't be as good without him.
I extend my sincere gratitude to our Head of Department, Dr. Dattaprasad A. Torse, and our college Principal, Dr. S. F. Patil, for their invaluable support during our project on "Enhanced Audio Noise Reduction System." Dr. Torse provided crucial guidance and encouragement, significantly improving our project with insightful advice. I appreciate the conducive learning environment fostered by both Dr. Torse and Dr. Patil, which allowed us to apply theoretical knowledge to real-life situations. Their leadership has been instrumental in shaping our academic journey, and I am thankful for their unwavering support and contributions to our project's success.

-The project team

ABSTRACT

The presence of background noise in audio recordings poses a significant challenge, degrading the overall quality and intelligibility of the recorded content. To address this issue, a noise reduction filter is proposed. The filter utilizes autoencoders, Wiener filtration, and Bandpass filter to effectively remove background noise while minimizing distortion of the desired signal. The filter's performance is evaluated using two audio files: one containing white noise and the other containing a radio signal with white noise. The results demonstrate that the filter effectively removes background noise, improving the clarity and intelligibility of the audio signal.

Contents

List of Tables

List of Figures

Chapter 1

Introduction

1.1 Motivation

Project Inspiration:

Background noise can be disruptive, affecting recordings in podcasts, lectures, music, and even home videos. Existing noise filters often fall short, either distorting voices or allowing unwanted noise to persist.

That's why we are developing a cutting-edge noise-busting filter. It acts like a smart shield, effectively blocking out unwanted sounds while preserving the clarity of voices and music. This endeavor goes beyond merely saving ears; it aims to enhance communication for everyone, from individuals with hearing impairments to those seeking a clear lecture experience. This project represents my opportunity to address a real-world problem and make a meaningful impact, one clean audio clip at a time.

1.2 Objectives

- Address the issue of background noise degradation in audio quality and intelligibility.
- Minimize distortion of the desired signal while removing background noise.
- Evaluate the effectiveness of autoencoders, Wiener filters, and bandpass filters for white noise removal.
- Determine the most efficient method based on Signal-to-Noise Ratio (SNR), Mean Squared Error (MSE), and Peak Signal-to-Noise Ratio (PSNR) metrics.
- Conclude that the best filter is the autoencoder due to its dynamic adaptability to new audio signals compared to other filtering techniques used.

1.3 Literature survey

Table 1.1: Literature Review

SI.No	Paper Title	Objective	Algorithm used	Limitations
1	[3]	auto encoders with downstream classifiers to enhance classification accuracy in a pipeline	Autoencoders	Potentially resulting in low flexibility.
2	[1]	This paper introduces noise absorption, minimizing distortion of speech while effectively reducing input signal noise	Recurrent Neural Network(RNNs)	Persistent sound noise, a long-standing issue, remains unresolved, posing a risk of audio file damage and impaired audio quality.
3	[2]	This paper aims to enhance decrypted speech quality for a satisfying auditory experience.	Wiener filtration	Results may signify the efficacy of our method in highly-distorted speech conditions.

1.4 Problem statement

Create a noise reduction filter that can remove background noise from audio recording.

1.5 Application in Societal Context

Improved Accessibility and Inclusion: By increasing the clarity and intelligibility of audio recordings, the filter can benefit people with hearing difficulties, allowing them to better access information and participate in conversations. This can be particularly crucial in educational settings, workplaces, and public spaces.

Environmental Benefits: In noisy environments like construction sites or transportation hubs, the filter can help reduce noise pollution, creating a more comfortable and healthy environment for residents and wildlife.

Enhanced Communication and Collaboration: In situations with background noise, effective communication can be hindered. This filter could improve communication clarity in various settings, such as video conferencing, interviews, and public announcements, leading to better collaboration and understanding.

1.6 Project Planning and bill of materials

Project Scope: The scope of this project is achieving improved audio quality, evaluating filter performance, and comparing the effectiveness of different approaches.

Technical Requirements: Technical requirements for this project include Python libraries such as `SciPy`, `sounddevice`, `soundfile`, `matplotlib`, `numpy`, `torch`, and `librosa`.

Project Steps:

1. Identify and collect the audio data for testing and training.
2. Preprocess the data, including noise addition for training the autoencoder if needed.
3. Implement the Wiener filter in Python using appropriate libraries.
4. Implement the bandpass filter in Python using DSP libraries (e.g., SciPy).
5. Implement the denoising autoencoder using a deep learning framework.
6. Evaluate the Wiener filter, bandpass filter, and autoencoder on test data.
7. Compare and analyze the results.

1.7 Organization of the report

In System Design:

- **Functional Block Diagram:** This is a visual representation of the major functional components of a system and their interconnections. It provides a high-level overview of how the system works on filtering the white noise. Each block represents a function or module, and arrows depict the flow of data or signals between them.
- **Design Alternatives:** In the design phase, we consider various options and alternatives for implementing each functional block. This involves evaluating different models such as Wiener filter, bandpass filter, Autoencoders, and approaches to meet the system requirements. Design alternatives help in selecting the most suitable solution based on factors like denoising ratio, performance, and scalability.
- **Final Design:** Once design alternatives are explored, the final design is chosen. This involves specifying the detailed configuration of each functional block. The final design takes into account all requirements and constraints and serves as the blueprint for the subsequent implementation phase.

Implementation Details:

- **Specifications and Final System Architecture:** Specified details about the requirements and constraints that the system must meet. The final system architecture provides a comprehensive view of how different components interact and work together to achieve the desired functionality.
- **Algorithm:** Algorithms are step-by-step procedures of how the filter works. In the context of system implementation, algorithms describe the logical flow of operations or computations required to achieve a particular task.
- **Flowchart:** A flowchart is a visual representation of the sequential flow of operations in a process or algorithm. It uses different shapes to represent different types of steps, such as input/output, processing, and decision points. Flowcharts help visualize the logical structure of an algorithm or system, making it easier to understand and implement.

Chapter 2

System design

2.1 Functional block diagram

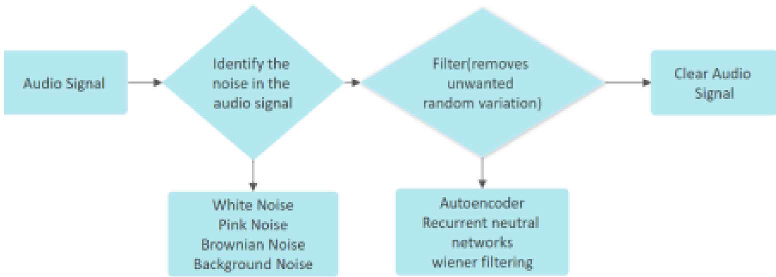

Figure 2.1: Block Diagram of denosing Audio

2.2 Design alternatives

1. Wiener Filter:

- Is a linear filter that makes an estimation of the original signal by minimizing the mean square error between the estimated signal and the original signal.
- The 'wiener' function is derived from the 'scipy.signal' module.
- The formula for the Wiener filter in your case is given by:

$$H(f) = \frac{X(f)}{X(f) + Y(f)}$$

where: H(f) is the Wiener filter transfer function.
X(f) is the power spectral density of the original audio signal.
Y(f) is the power spectral density of the noise.

2. Bandpass Filter:

- A bandpass filter is applied to the noisy audio signal to individually allow for a certain range of frequencies to pass through them, efficiently removing unwanted frequencies.
- The transfer function is given by:

$$X(f) = \frac{Y(f)}{(1 + (\frac{F}{Fc})^{2n})}$$

where: X(f) is the transfer function.
Y(f) is the gain.
F is the frequency, and Fc is the center frequency.
n is the filter order.

- The bandpass filter uses a Butterworth filter with the gain given by:

$$Y(f) = \frac{1}{\sqrt{1 + (\frac{F}{Fc})^{2n}}}$$

where: Y(f) is the Butterworth filter gain.
F is the frequency, and Fc is the cut-off frequency.

- The function 'signal.lfilter' is used to apply the bandpass filter.

3. Denoising Autoencoders:

- In denoising autoencoders, we add white noise to the clean audio signal to create a noisy audio file, using the function 'add_white_noise', whose mathematical expression is:

$$noisy\_audio\_signal = clean\_audio\_signal + white\_noise\_level \times random\_noise\_level$$

- The model used in our project is a Convolutional Neural Network (CNN)-based denoising autoencoder model. The mathematical expression used is:

$$denoised\_output = model(noisy\_audio\_reshaped)$$

2.3 Final design

Working:

- The **Wiener filter** is a **linear filter** that minimizes the mean square error between the estimated signal and the original signal.
- It is commonly used for **signal enhancement** in the presence of additive noise.

Ease of Implementation:

- Wiener filters are **relatively straightforward to implement**, especially in cases where the statistical properties of the signal and noise are well understood and standard functions available in many signal processing libraries.

Chapter 3

Implementation details

3.1 Specifications and final system architecture

Specifications for Wiener Filter

Input:

- Noisy Audio Signal: The audio signal contaminated with white noise.
- Noise Level: A parameter indicating the level of white noise in the input audio.

Processing Steps:

- Wiener Filter Application: Apply the Wiener filter to the noisy audio signal.
- Use the Wiener filter implementation, such as `scipy.signal.wiener`, to estimate the clean signal from the noisy signal.
- The `mysize` parameter may be adjusted based on the characteristics of the noise and signal.
- The noise level parameter (`noise_level`) is used in the Wiener filter.

Output:

- Cleaned Audio Signal: The audio signal after the application of the Wiener filter.

Usage:

- The user provides the path to the noisy audio file and sets the noise level parameter.
- The Wiener filter function (`remove_white_noise`) is called with the noisy audio and noise level as inputs.

Dependencies:

- `scipy.signal.wiener` for the Wiener filter implementation.
- `numpy` for numerical operations.

Adjustable Parameters:

- `mysize`: Parameter in the Wiener filter function, representing the size of the local region used for estimation.

- `noise_level`: Parameter indicating the level of white noise in the input audio.

Visualization:

- The waveforms of the original noisy audio and the cleaned audio after the Wiener filter may be plotted using the `plot_audio_waveform` function.

3.2 Algorithm

Load Noisy and Clean Audio:

- **Load the noisy audio** from the specified file path (`noisy_audio_path`).
- **Load the original (clean) audio** from the specified file path (`clean_audio`).

Set Parameters:

- **Set the noise level** (`noise_level`) used for adding white noise. This parameter will also be used for the Wiener filter.

Apply Wiener Filter:

- **Call the `remove_white_noise` function** with the noisy audio and the noise level as parameters.
- The `remove_white_noise` **function** uses the `wiener` function from `scipy.signal` to apply the Wiener filter to the noisy audio.
- The **filtered/cleaned audio is returned**.

Play Original Noisy Audio:

- **Print a message** indicating that the noisy audio with white noise is about to be played.
- **Call the `play_audio` function** to play the original noisy audio.

Play Cleaned Audio after Wiener Filter:

- **Print a message** indicating that the cleaned audio after the Wiener filter is about to be played.
- **Call the `play_audio` function** to play the cleaned audio after applying the Wiener filter.

Save Cleaned Audio to File:

- **Specify the output file path** (`output_path`) **for the cleaned audio after the Wiener filter**.
- **Call the `save_audio` function** to save the cleaned audio to the specified file path.

Display Waveforms:

- **Plot the waveforms** of the original noisy audio and the cleaned audio after the Wiener filter.
- **Use the `plot_audio_waveform` function** for visualization.

3.3 Flowchart

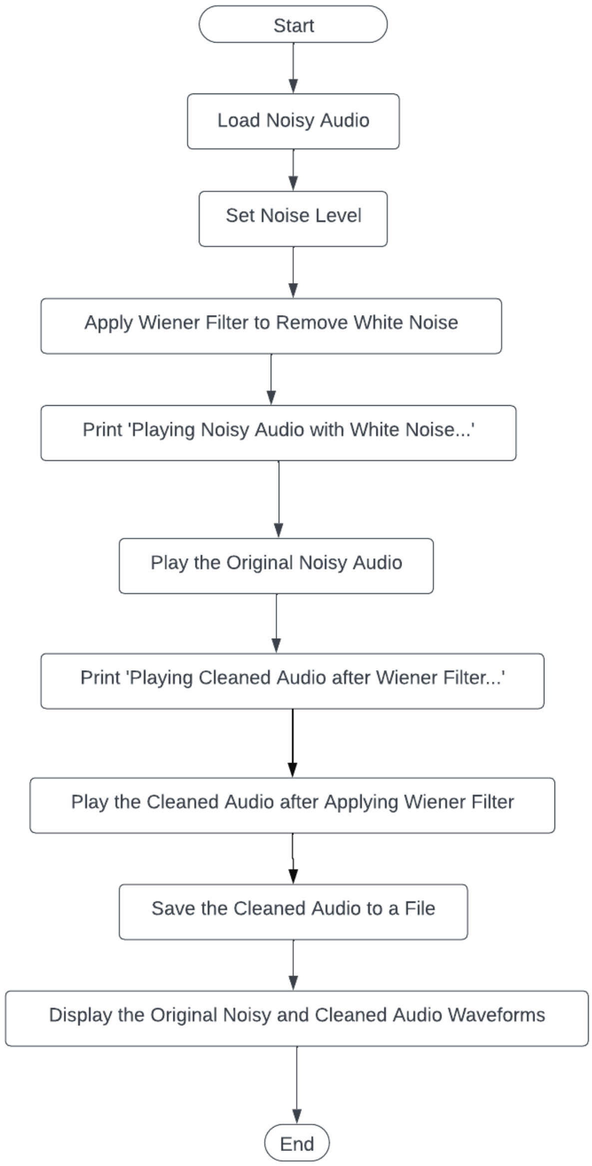

Figure 3.1: Flowchart of Wiener filter

Chapter 4

Optimization

4.1 Introduction to optimization

To optimize the results obtained, our goal is to enhance Wiener filter results. We achieve this by comparing signal-to-noise ratio (SNR), mean squared error (MSE), and peak signal-to-noise ratio (PSNR) values with those from two other models—the denoising autoencoder and bandpass filter. This straightforward analysis helps us identify which method works best in improving signal quality and minimizing noise in audio processing. Our aim is to streamline and enhance audio applications by selecting the most effective filtering technique based on these performance metrics.

4.2 Types of Optimization

- **Bandpass filtering**

  White noise has a characteristic of having a flat frequency spectrum. Bandpass filtering is a well-established and efficient DSP technique that allows you to selectively pass the frequency of the signal while attenuating frequencies outside that range (white noise).

- **Denoising Autoencoder**

  Denoising autoencoder is beneficial. It is a type of neural network that is trained to reconstruct clean signals from noisy inputs. The model learns to capture patterns and structures in data, which can be helpful for denoising.

4.3 Selection and justification of optimization method

Bandpass filter:

- **Simplicity:** Bandpass filters are relatively simple to design and implement, making them accessible for various applications. They consist of a combination of low-pass and high-pass filters, which are simpler than more complex filter types.
- **Effectiveness:** Bandpass filters efficiently isolate and pass a specific range of frequencies while attenuating those outside the desired band. This targeted frequency response makes them highly effective for applications where isolating a particular signal frequency or range is crucial.

Denoising autoencoder:

- **Adaptability:** Denoising autoencoders are adaptable to various types of noisy input data. They can effectively learn and adapt to the specific characteristics and patterns present in different types of noise, allowing them to handle diverse and complex sources of noise in the input signals. The adaptability of denoising autoencoders makes them versatile for tasks of audio signal restoration or any application where noisy data needs to be transformed into cleaner representations.
- **Non-linearity:** Denoising autoencoders introduce non-linearity into the learning process, enabling them to capture complex relationships and structures within the data. The non-linear activation functions used in the neural network layers allow the model to learn intricate features and representations that may have non-linear dependencies.

Chapter 5

Results and discussions

5.1 Result Analysis

As a Results to select the best suitable method to filter the white noise is by comparing the results of each model. We have plotted the audio signals, where Time (s) is defined on x-axis and Amplitude on y-axis for all the plotted figures below. Fig 5.1: clean audio signal, Fig 5.2: White noise signal, Fig 5.3: Noisy audio signal, Fig5.4: applied Wiener Filtered, Fig 5.5 Bandpass Filtered and Fig 5.6 Autoencoder Denoised. We compare the models based on their values of Signal-to-Noise Ratio (SNR), Mean Squared Error (MSE), and Peak Signal-to-Noise Ratio (PSNR). For the following three parameters, Fig 5.9: shows Bar graph of SNR, Fig 5.8: Bar graph of MSE and Fig 5.7: Bar graph of PSNR.

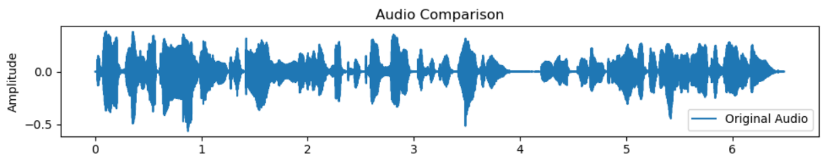

Figure 5.1: Clear Audio Signal

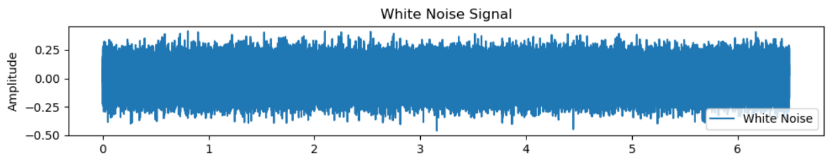

Figure 5.2: White Noise Signal

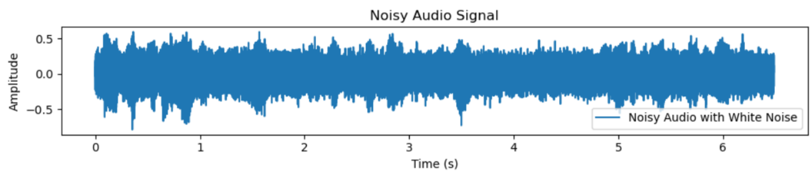

Figure 5.3: Noisy Audio Signal

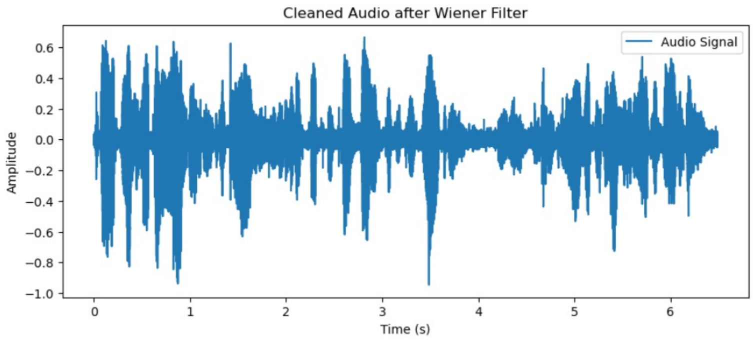

Figure 5.4: Wiener Filtered Audio Signal

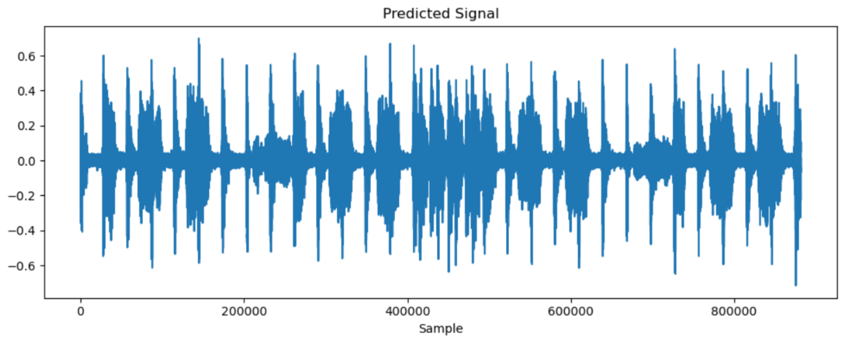

Figure 5.5: Autoencoder Filtered Signal

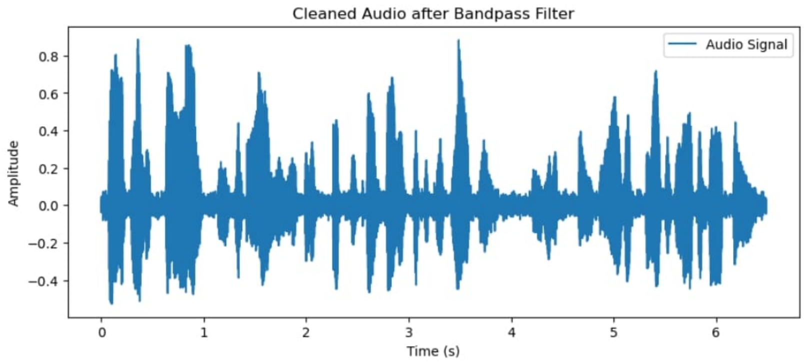

Figure 5.6: Bandpass Filtered Audio Signal

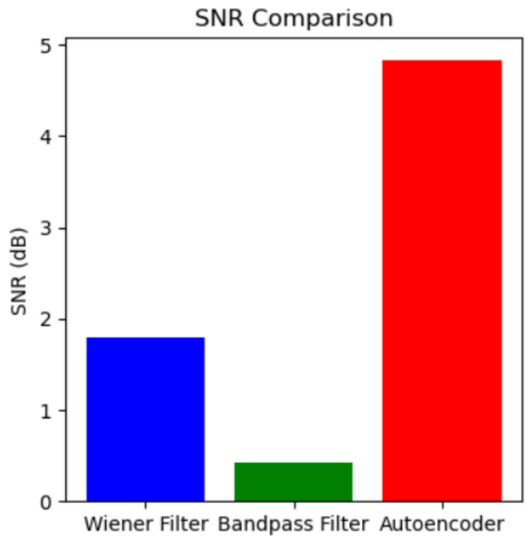

Figure 5.7: SNR Comparison

1) Signal to Noise Ratio (SNR)

$$SNR = 10 \cdot \log_{10}\left(\frac{\sum clean\_audio^2}{\sum(clean\_audio - cleaned\_audio)^2}\right)$$

where: clean_audio = Original clean audio signal.
noisy_audio = Noisy audio signal (with added white noise or other disturbances).
cleaned_audio = Denoised audio signal obtained from the respective denoising method.

- Wiener Filter = 1.7946801824557168 dB
- Bandpass Filter = 0.42112599168153797 dB
- Autoencoder = 4.834953546524048 dB

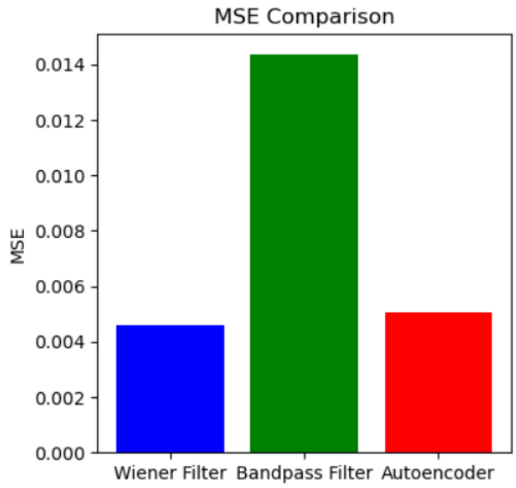

Figure 5.8: MSE Comparison

2) Mean Squared Error (MSE)
Mean Squared Error (MSE) is calculated using:

$$MSE = \frac{1}{n}\sum_{i=1}^{N}(CLEAN\_audio[i] - cleaned\_audio[i])^2$$

where: CLEAN_audio = Original clean audio signal.
cleaned_audio = Denoised audio signal obtained from the respective denoising method.

- Wiener Filter = 0.004614444125793488
- Bandpass Filter = 0.014371632528890113
- Autoencoder = 0.005043465178459883

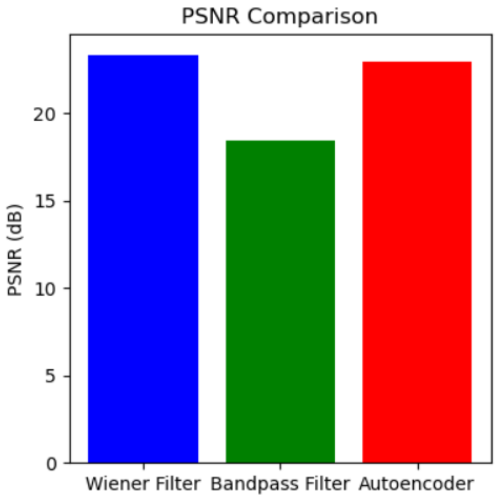

Figure 5.9: PSNR

3) Peak Signal to Noise Ratio (PSNR)
PSNR is calculated using:

$$PSNR = 10 \cdot \log_{10}\left(\frac{max\_intensity^2}{MSE}\right)$$

where: clean_audio = Original clean audio signal.
cleaned_audio = Denoised audio signal obtained from the respective denoising method.
max_intensity = Maximum intensity of the audio signal (assumed to be 1.0 in this case).

- Wiener Filter = 23.358806083197 dB
- Bandpass Filter = 18.42493895886785 dB
- Autoencoder = 22.97270973319909 dB

5.2 Discussion on optimization

Table 5.1: Results

SI.No	Methods	SNR	MSE	PSNR
1	Wiener Filter	1.79dB	0.0046	23.35dB
2	Autoencoder	4.083dB	0.0050	22.97dB
3	Bandpass Filter	0.42dB	0.0143	18.424dB

- In Table 5.1, accurate values for three parameters used to compare the three models are presented. From the plots and the tables, it is evident that the autoencoder has a higher Signal to Noise Ratio (SNR) (4.083 dB) and Peak Signal to Noise Ratio (PSNR) (22.97 dB) with a lower Mean Squared Error (MSE) (0.050) than the other two models used.
- An increased SNR value in a model indicates reduced noise in the filtered audio signal. When it comes to MSE, the model having the least value indicates filtered audio is close to the original signal. For PSNR, it is quite like SNR but is given in decibels; a higher PSNR value indicates better signal quality.
- As a result, the autoencoder performs the best, followed by the Wiener filter and the bandpass filter, which are the least performing filters when it comes to filtering white noise.

Chapter 6

Conclusion and future Scope

6.1 Conclusion

In this paper, we have discussed how the wiener filter, bandpass filter, and autoencoder can be used to filter out white noise and which method is the best to filter out the noise from an audio signal. We have compared the efficiency of the models based on Signal-to-Noise Ratio (SNR), Mean Squared Error (MSE), and Peak Signal-to-Noise Ratio (PSNR). After comparing the three models, Autoencoder gives the best results as compared to the Wiener filter and bandpass filter. Using an autoencoder is beneficial because it is more dynamic in adapting to new audio signals as compared to the other filtering techniques used.

6.2 Future Scope

In the future, there is a keen interest in enhancing the efficiency of autoencoders for audio noise reduction. Further exploration will focus on their efficacy in mitigating various noise types, including Brownian and pink noise. This advancement holds potential benefits for societal applications, such as enhancing audio quality in communication systems and entertainment, ensuring a clearer and more enjoyable auditory experience for users.

Bibliography

[1] Akarsh S. M, Rajashekar Biradar, and Prashanth V Joshi. Implementation of an active noise cancellation technique using deep learning. In *2022 International Conference on Inventive Computation Technologies (ICICT)*, pages 249–253, 2022.

[2] Hilal H. Nuha and Ahmad Abo Absa. Noise reduction and speech enhancement using wiener filter. In *2022 International Conference on Data Science and Its Applications (ICoDSA)*, pages 177–180, 2022.

[3] Shaikh Akib Shahriyar, M. A. H. Akhand, N. Siddique, and T. Shimamura. Speech enhancement using convolutional denoising autoencoder. In *2019 International Conference on Electrical, Computer and Communication Engineering (ECCE)*, pages 1–5, 2019.