VISVESVARAYA TECHNOLOGICAL UNIVERSITY

"Jnana Sangama", Belagavi-590018

A

Project Synopsis

On

"Advanced Audio Enhancement System"

SUBMITTED IN PARTIAL FULFILLMENT FOR 6TH SEMESTER

BACHELOR OF ENGINEERING

IN

COMPUTER SCIENCE AND ENGINEERING

SUBMITTED BY

SRISTI PRIYA	(1JB22CS160)
SNEHA MAL	(1JB22CS152)
SURATH CHOUDHARY	(1JB22CS167)
SAMEER CHOUDHURY	(1JB22CS133)

UNDER THE GUIDANCE OF

Dr. Roopa M J
Associate Professor
Dept. of CSE, SJBIT

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

SJB INSTITUTE OF TECHNOLOGY

No.67, BGS Health& Education City, Dr. Vishnuvardhan Rd, Kengeri, Bengaluru, Karnataka 560060
An Autonomous Institute under VTU
Approved by AICTE - New Delhi, Accredited by NAAC A+, Accredited by NBA

2024 – 2025

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

CHAPTER 1

INTRODUCTION

Audio compression has become a crucial aspect of digital media processing, enabling efficient storage and transmission of audio data while preserving its quality. Traditional audio compression methods are broadly classified into two categories: lossless and lossy compression. Lossless compression techniques, such as FLAC, WavPack, and Monkey’s Audio, focus on reducing file size without any loss in audio fidelity. However, these methods often struggle to achieve compression rates comparable to modern lossy encoders like MP3, which prioritize aggressive data reduction at the cost of slight quality loss. Consequently, developing a lossless audio compression technique that achieves a higher compression ratio without compromising audio quality remains a significant challenge.

To address this, the proposed research introduces an innovative artificial neural network (ANN)-based lossless audio encoder that utilizes dynamic data segregation and adaptive transformations. This novel approach aims to enhance compression efficiency by grouping and dynamically distributing audio data during encoding. The method employs an ANN model combined with Huffman encoding to achieve superior compression performance. By encoding five audio data points into four elements in each hidden layer, the system achieves approximately 20% compression per layer. With eight hidden layers, the overall compression ratio surpasses traditional lossless methods, achieving an impressive 85% average compression rate across multiple audio genres.

Despite its strengths, the proposed method faces certain challenges. Increased computational complexity and longer encoding/decoding times pose limitations for real-time applications. Additionally, while the model achieves improved compression performance, slight reductions in PSNR values have been observed under certain conditions. Future enhancements will focus on optimizing the model for faster processing speeds, improved PSNR outcomes, and broader dataset adaptability.

CHAPTER 2

LITERATURE SURVEY

Table 2.1 Summary of Literature Papers

1.	Milling M, Liu S, Triantafyllopoulos A et al.	Audio Enhancement for Computer Audition—An Iterative Training Paradigm Using Sample Importance 2024	Proposed an iterative training approach where sample importance is dynamically adjusted to improve model learning.	Advantage: Improved model robustness Disadvantage May require large datasets for optimal performance
2.	Haohe Liu, Ke Chen, Qiao Tian, Wenwu Wang, Mark D. Plumbley	AUDIOSR: Versatile Audio Super-Resolution at Scale 2024	Deep learning-based audio super-resolution model for enhancing low-quality audio at scale	Advantage:High scalability and effectiveness in various domains Disadvantage: May struggle with extreme low-quality input
3.	Manuel Milling et al.	Audio Enhancement for Computer Audition—An Iterative Training Paradigm (2024)	Utilized a U-Net architecture for audio enhancement with iterative optimization techniques. Focused on improving performance in speech command recognition, automatic speech recognition (ASR), speech emotion recognition (SER), and acoustic scene classification (ASC).	**Advantages:** Enhanced noise suppression and improved performance in audio recognition tasks. **Disadvantages:** Limited generalization to clean data when trained with noisy data .

4.	Seungkwon Beack, Byeongho Jo, Wootaek Lim, Jungwon Kang	Audio Compression Technique for Low Delay and High Efficiency Using Complex Audio Data (2024)	Proposed an audio compression method using complex frequency-domain data to improve latency and maintain high audio quality. Utilized quantization techniques like Modulated Complex Lapped Transform (MCLT) and Discrete Fourier Transform (DFT) for efficient compression.	**Advantages:** Achieves low latency (<50ms) with MOS 4.5 quality at 96 kbps. Improves compression ratios without sacrificing quality. **Disadvantages:** The system's complexity may increase computational requirements, limiting its application on low-performance devices.
5.	Haohe Liu, Ke Chen, Qiao Tian, Wenwu Wang, Mark D. Plumbley	AudioSR: Versatile Audio Super-Resolution at Scale (2024)	Proposed a diffusion-based generative model called AudioSR that performs audio super-resolution across versatile audio types. The method uses a latent diffusion model combined with a neural vocoder to reconstruct high-frequency audio data effectively.	Advantages**:** Capable of upsampling input audio from 2 kHz to 16 kHz to 24 kHz bandwidth with a 48 kHz sampling rate. Demonstrated robust results across speech, music, and sound effects. Disadvantages**:** Increased computational demand may limit its performance on low-end hardware, and the model may require additional fine-tuning for specific audio domains.

CHAPTER 3

CHALLENGES

1. **Noise Reduction Complexity:** Implementing effective noise reduction algorithms using Python libraries like noisereduce or scipy can be challenging, especially when dealing with diverse audio environments containing variable noise patterns.
2. **Latency and Processing Speed:** Real-time audio enhancement often demands efficient algorithms. Achieving low latency while ensuring high-quality output using Python libraries can require significant optimization.
3. **Balancing Quality and Performance:** Enhancing audio clarity often involves trade-offs. Techniques such as equalization and dynamic range compression may improve sound quality but increase processing overhead.
4. **Compatibility Issues:** Ensuring compatibility with various audio formats (e.g., WAV, MP3, FLAC) and sample rates can be complex when integrating libraries like librosa, soundfile, or pydub.
5. **Resource Consumption:** Advanced audio enhancement techniques, especially those involving machine learning models like torch or openai-whisper, can be resource-intensive, requiring substantial memory and processing power.
6. **Feature Engineering:** Developing custom filters, equalizers, and dynamic range controllers demands a strong understanding of signal processing techniques and careful tuning to avoid audio distortion.
7. **User Interface Development:** Building a user-friendly interface for non-technical users, ensuring seamless audio file uploads, enhancement settings, and file exports can add development complexity.
8. **Error Handling and Robustness:** Ensuring the application can handle corrupt audio files, unsupported formats, or incomplete data is essential for reliability.

CHAPTER 4

MOTIVATION

- **Improved Audio Clarity:** Enhancing audio quality for podcasts, interviews, and music production to provide a richer listening experience.
- **Noise Reduction in Recordings:** Developing tools to filter out background noise in home recordings, field interviews, or old audio files.
- **Accessibility for Content Creators:** Providing affordable and accessible solutions for independent content creators, streamers, and educators who require clear audio without expensive equipment.
- **Automation in Audio Processing**: Motivated by the need to reduce manual adjustments by implementing automated enhancement techniques using Python modules.
- **Integration with Modern Technologies:** Enabling seamless integration with machine learning models for adaptive noise reduction and intelligent sound enhancement.
- **Versatility Across Applications:** Addressing the demand for enhanced audio quality in industries such as media production, gaming, telecommunication, and forensics.
- **Cost-Effective Solutions:** Creating efficient audio enhancement tools using open-source Python libraries to reduce dependency on expensive proprietary software.
- **Research and Innovation:** Encouraging exploration of novel methods in digital signal processing, machine learning models, and adaptive algorithms for improved audio enhancement solutions.
- **User-Centric Design:** Developing user-friendly intefaces and workflows to ensure ease of use for both beginners and professional audio engineers.
- **Support for Remote Collaboration:** Facilitating clear audio communication in remote work environments, virtual meetings, and online learning platforms by improving audio clarity and reducing noise.

CHAPTER 5

PROBLEM STATEMENT AND OBJECTIVES

5.1PROBLEM STATEMENT

- **Audio Quality Degradation**: Audio recordings often suffer from poor sound clarity, background noise, and uneven volume levels, especially in non-professional setups such as home recordings, interviews, and conference calls. Environmental factors, such as echoes, wind noise, and microphone limitations, can significantly degrade audio quality. This issue is particularly prominent in industries like content creation, broadcasting, and education, where clear audio is essential. Developing an effective solution that can automatically enhance audio clarity, reduce noise, and balance sound levels using Python modules is a crucial challenge.
- **Limited Accessible Tools**: While various professional audio enhancement tools exist, they are often expensive, require specialized knowledge to operate, or demand significant manual effort to fine-tune. This creates a barrier for independent content creators, small businesses, and casual users seeking high-quality audio enhancement. An accessible, cost-effective, and user-friendly Python-based solution that leverages automation and advanced algorithms is essential to address this gap.

5.2 OBJECTIVES

- Enhanced Noise Reduction: Develop algorithms using Python libraries like noisereduce and scipy to effectively minimize background noise without distorting key audio components.
- Improved Audio Clarity and Volume Control: Implement equalization (EQ), dynamic range compression, and volume normalization techniques to achieve balanced and clear audio output.
- User-Friendly Interface: Design a simple, intuitive interface that allows users to enhance audio files efficiently, ensuring accessibility for both technical and non-technical users.

CHAPTER 6

METHODOLOGY

1. **Pre-Processing:**

   Apply Python libraries like librosa and scipy for loading, sampling, and normalizing audio data to ensure consistency in input formats, frequencies, and amplitudes.

2. **Noise Reduction Algorithm:**

   Implement advanced noise reduction techniques using noise reduce to minimize unwanted sounds like background chatter, static, or environmental noise while preserving key audio features.

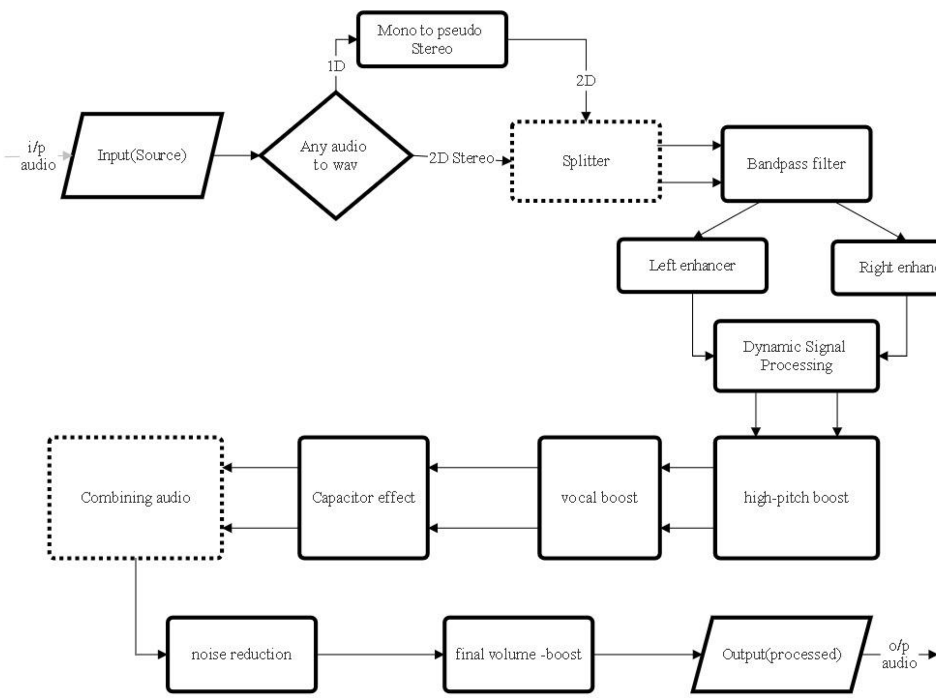

Fig 6.1 Block diagram of “Advanced Audio Enhancement System”

3. Equalization (EQ):

Develop frequency adjustment methods using Python's scipy and librosa to enhance vocal clarity, boost essential frequencies for clearer dialogue, and reduce harsh tones or muffled sound.

4. Dynamic Range Compression:

Introduce compression techniques that balance loud and quiet audio parts to ensure smoother output by controlling sound peaks and enhancing softer sounds.

5. Volume Normalization:

Implement normalization algorithms to ensure consistent audio loudness across recordings, preventing sudden volume spikes or drops during playback.

6. Audio Effects Implementation:

Add optional effects like bass boost for deeper sound, reverb control for spatial enhancement, and echo reduction for clearer speech and enhanced music quality.

7. User Interface Development:

Design an intuitive interface using frameworks like Tkinter or PyQt that provides interactive controls, allowing users to apply effects, adjust settings, and export enhanced audio files.

8. Testing and Validation:

Perform extensive testing using metrics such as Mean Squared Error (MSE), Peak Signal-to-Noise Ratio (PSNR), and user feedback to evaluate performance, ensuring improved clarity, reduced noise, and consistent loudness across samples.

CHAPTER 7

POSSIBLE OUTCOMES

1. **Enhanced Audio Quality:**
   The tool is expected to produce clearer, distortion-free audio by effectively reducing noise and improving frequency balance.

2. **Improved Speech Intelligibility:**
   Enhanced vocal clarity will ensure better communication quality, particularly in podcasts, lectures, and online meetings.

3. **User-Friendly Interface:**
   The developed solution will feature an intuitive interface that allows non-technical users to easily enhance audio without complex configurations.

4. **Automation and Efficiency:**
   The automated processing steps will minimize manual intervention, improving workflow efficiency for content creators and professionals.

5. **Wide Application Support:**
   The tool will be compatible with various audio formats such as WAV, MP3, and FLAC, catering to a diverse range of use cases.

6. **Real-Time Enhancement Potential:**
   With further optimizations, the solution may enable near real-time audio enhancement capabilities for live streaming and conferencing.

7. **Resource Efficiency:**
   The solution will aim to balance computational demands, ensuring optimal performance on a wide range of hardware systems.

REFERENCES

[1] Haohe Liu, Ke Chen, Qiao Tian, Wenwu Wang, Mark D. Plumbley. (2023). AudioSR: Versatile Audio Super-resolution at Scale. Retrieved from AudioSR: Versatile Audio Super-resolution at Scale | arXiv

[2] **Isabella Lenz, Yu Rong, Daniel Bliss, Julie Liss, Visar Berisha**. (2025). *A Speech Production Model for Radar: Connecting Speech Acoustics with Radar-Measured Vibrations*. Retrieved from arXiv:2503.15627

[3] **Luca Jiang-Tao Yu, Running Zhao, Sijie Ji, Edith C. H. Ngai, Chenshu Wu**. (2024). *USpeech: Ultrasound-Enhanced Speech with Minimal Human Effort via Cross-Modal Synthesis*. Retrieved from arXiv:2410.22076

[4] **Manuel Milling, Shuo Liu, Andreas Triantafyllopoulos, Ilhan Aslan, Björn W. Schuller**. (2024). *Audio Enhancement for Computer Audition—An Iterative Training Paradigm Using Sample Importance*. Retrieved from arXiv:2408.06264

[5] **Junan Zhang, Jing Yang, Zihao Fang, Yuancheng Wang, Zehua Zhang, Zhuo Wang, Fan Fan, Zhizheng Wu**. (2025). *AnyEnhance: A Unified Generative Model with Prompt-Guidance and Self-Critic for Voice Enhancement*. Retrieved from arXiv:2501.15417

[6] **Haohe Liu, Ke Chen, Qiao Tian, Wenwu Wang, Mark D. Plumbley**. (2023). *AudioSR: Versatile Audio Super-Resolution at Scale*. Retrieved from arXiv:2309.07314

[7] **Wilfredo J. Robinson M., Medhani Kalal**. (2024). *Spatial Audio-Enhanced Multimodal Graph Rendering for Efficient Data Trend Learning on Touchscreen Devices*. Retrieved from NSF Public Access

[8] **Keren Shao, Ke Chen, Shlomo Dubnov**. (2024). *Music Enhancement with Deep Filters: A Technical Report for The ICASSP 2024 Cadenza Challenge*. Retrieved from arXiv:2404.11116

[9] **Haoyang Li, Jia Qi Yip, Tianyu Fan, Eng Siong Chng**. (2025). *Speech Enhancement Using Continuous Embeddings of Neural Audio Codec*. Retrieved from arXiv:2502.16240

[10] **Tianrui Wang, Jin Li, Ziyang Ma, Rui Cao, Xie Chen, Longbiao Wang, Meng Ge, Xiaobao Wang, Yuguang Wang, Jianwu Dang, Nyima Tashi**. (2024). *Progressive*