Speech Enhancement

Using U-Net Architecture

Mini-Project report submitted in partial fulfillment of the
Requirements for the degree of

Master of Engineering
ME (Big Data Analytics)

By

Nikhil M (241058020)

Nikhil S G (241058024)

Vinayashree M Shet (241058030)

MANIPAL
ACADEMY of HIGHER EDUCATION
(Deemed to be University under Section 3 of the UGC Act, 1956)

MANIPAL SCHOOL OF INFORMATION SCIENCES
(A Constituent unit of MAHE, Manipal)

Acknowledgement

It is great pleasure to thank the people behind the success of this Mini-Project activity. I owe my deepest gratitude to all those who guided, inspired, and helped me to complete this project.

I would like to take this opportunity to express my gratitude and heartiest thanks to my panel members, **Prof. Mr. Raghudathesh G P**, Associate Professor, Manipal School of Information Sciences for his inspirational support and guidance throughout my project period.

My heartful gratitude to **Dr Keerthana Prasad** Professor & Director Manipal School of Information Sciences for his full support and encouragement during the Mini-Project activity.

I would like to thank all sources mentioned in the references.

LIST OF FIGURES

Contents

Abstract

A speech enhancement system based on U-Net architecture is proposed to improve audio clarity by attenuating environmental noise. The system uses Short-Time Fourier Transform (STFT) magnitude spectrograms of noisy speech as input, allowing the U-Net model to predict and isolate noise patterns, which are subsequently removed from the original spectrogram to preserve the clean voice signal. With symmetric skip connections, the U-Net architecture retains essential spatial details for accurate noise reduction. Training data includes clean speech samples from the LibriSpeech dataset, combined with environmental noises from the ESC-50 and SiSec datasets. The final denoised audio is reconstructed by combining the predicted clean spectrogram with the original phase information, resulting in a clearer, more intelligible output. Experimental results demonstrate the system's generalization across various noise types, including alarms, engines, and ambient sounds, leading to significant improvements in audio quality. This approach offers a robust solution for enhancing speech clarity in noisy environments, with applications in telecommunications, assistive devices, and media production.

Chapter 1
Introduction

This project is dedicated to developing a cutting-edge speech enhancement system that enhances audio clarity by efficiently reducing environmental noise. It utilizes magnitude spectrograms, which convert audio signals into 2D images showing time and frequency, and processes them through a U-Net deep learning model. The model is trained to identify and remove noise from voice spectrograms, resulting in much clearer audio. Training involves clean speech from the LibriSpeech dataset and diverse noise samples from the ESC-50 and SiSec datasets, using advanced data augmentation techniques and GPU optimization to manage various noise environments. The enhanced audio is then reconstructed by combining the cleaned spectrogram with the original phase. This system has versatile applications, including improving communication quality in telecommunications, enhancing audio for assistive devices, and refining sound in media production, making it a valuable tool for various professional and consumer uses.

Chapter 2
Literature Survey

2.1 U-Net: Convolutional Networks for Biomedical Image Segmentation

Abstract:

The U-Net architecture was developed to address the unique challenges in biomedical image segmentation, such as the need to precisely identify and differentiate cellular structures in images, often with small training datasets. Unlike standard convolutional neural networks (CNNs), which were typically used for classification, U-Net was structured to handle segmentation tasks, where pixel-level classification is needed.

Methodology:

The methodology of the *U-Net* architecture focuses on creating a symmetric encoder-decoder structure specifically designed for segmentation tasks, primarily in biomedical imaging. The architecture consists of a contracting path, also known as the encoder, and an expansive path, or decoder. In the contracting path, the model applies a series of convolutional and max-pooling layers to progressively capture and condense spatial information, resulting in deeper feature maps that encode the context of the image. Following this, the expansive path, or decoder, aims to recover spatial resolution through upsampling layers, reconstructing the image's segmentation details

Outcome:

The U-Net model, introduced by Olaf Ronneberger, Philipp Fischer, and Thomas Brox, is a groundbreaking advancement in the field of biomedical image segmentation. The primary motivation behind U-Net was to address the challenges in segmenting complex medical images, a task essential for diagnostic and treatment planning in medical fields. Traditional segmentation models struggled with precise boundary

2.2 Singing voice separation with deep U-Net convolutional networks

Abstract:

The paper "Singing Voice Separation with Deep U-Net Convolutional Networks" by Jansson et al. (2017) presents a deep learning approach using U-Net architectures to separate singing voices from music accompaniment in audio tracks. This work is part of the International Conference on Music Information Retrieval (ISMIR) and represents a significant step forward in music source separation.

Methodology:

The authors adapted the U-Net convolutional neural network, initially used in biomedical image segmentation, for the purpose of audio source separation. In their approach, spectrograms of music tracks (a visual representation of audio frequencies over time) serve as inputs to the U-Net model. The U-Net architecture is designed to capture both low-level and high-level audio features through its encoder-decoder structure and its unique skip connections, which allow information from earlier layers to inform later layers in the network.

Outcome:

The experiments demonstrated that U-Net could effectively isolate vocals from music tracks, outperforming previous methods in terms of both quality and intelligibility of the separated vocals. The separated singing voices retained significant clarity and minimized interference from background music, proving the -Net model's capability in audio signal processing.

2.3 U-Net and Its Variants for Medical Image Segmentation: A Review of Theory and Applications

Abstract:

The paper titled "U-Net and Its Variants for Medical Image Segmentation: A Review of Theory and Applications" by N. Siddique, S. Paheding, C. P. Elkin, and V. Devabhaktuni, published in *IEEE Access* in 2021, provides a comprehensive review of the U-Net model and its variations, focusing on their applications in medical image segmentation.

Methodology:

The U-Net architecture, renowned for its encoder-decoder structure with skip connections, was designed to retain important spatial information while capturing detailed features within images. The encoder portion of U-Net reduces the spatial dimensions through a series of convolutional and pooling layers, extracting features progressively. This is followed by a decoder network, which reconstructs the image resolution through upsampling layers. Skip connections bridge each encoder layer to the corresponding decoder layer, allowing the network to retain high-resolution features that would otherwise be lost. This architecture has made U-Net exceptionally well-suited for applications requiring precise boundary detection, such as organ and tissue segmentation in medical images.

Outcome:

The review demonstrates that U-Net and its numerous variants have led to significant improvements in medical image segmentation accuracy. The adoption of U-Net models has facilitated advancements in various areas, including tumor detection, organ segmentation, vessel structure delineation, and lesion identification, thereby contributing to better diagnostic and treatment processes in healthcare. For instance, accurate tumor segmentation enables precise measurements for monitoring growth and treatment response, while organ segmentation helps in planning radiation therapy by accurately targeting the treatment area.

Chapter 3

Methodology

3.1 U-Net architecture

The U-Net architecture is a popular neural network design for image segmentation tasks, and it's known for its efficiency and effectiveness in capturing both high-level and low-level features in images. Here's a detailed breakdown of its architecture and workflow:

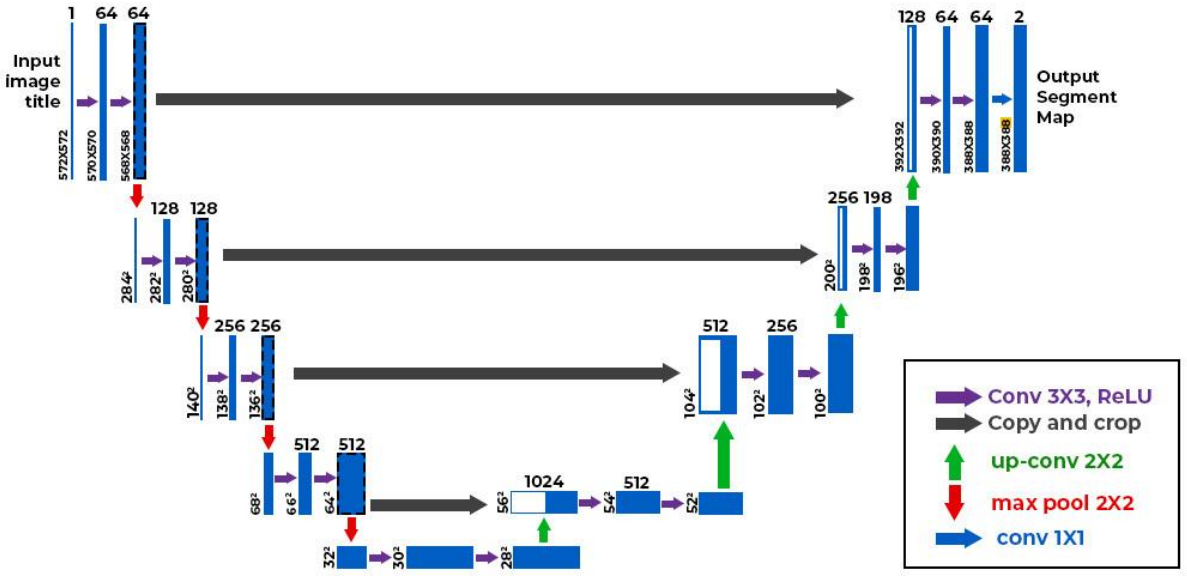

Fig.3.1 U-Net Architecture

1. Contracting Path (Encoder):

- Downsampling Blocks: The contracting path consists of a series of convolutional layers followed by max-pooling layers. Each block typically includes two convolutional layers (with small 3x3 filters) followed by a ReLU activation function.
- Pooling: After the convolutions, a max-pooling layer (usually with a 2x2 filter) reduces the spatial dimensions of the feature maps. This process is repeated several times, allowing the network to capture high-level features and reduce the dimensionality of the data.

2. Bottleneck:

- Bridge: At the bottom of the contracting path, there's a bottleneck layer where the feature maps are at their smallest size. This part usually consists of several convolutional layers, helping the network capture the most abstract features**.**

3. Expansive Path (Decoder):

- Upsampling Blocks: The expansive path mirrors the contracting path. It involves upsampling layers (using transposed convolutions or upsampling followed by convolutions) that increase the spatial dimensions of the feature maps.
- Skip Connections: The key feature of U-Net is the use of skip connections, which concatenate the feature maps from the contracting path with those from the corresponding layer in the expansive path. This helps the network retain spatial information that might otherwise be lost during downsampling**.**

4. Output Layer:

- Final Convolutions: After the upsampling, the final layer is a 1x1 convolution that reduces the number of channels to the desired output size (e.g., the number of classes in segmentation tasks).

3.2 Model Workflow

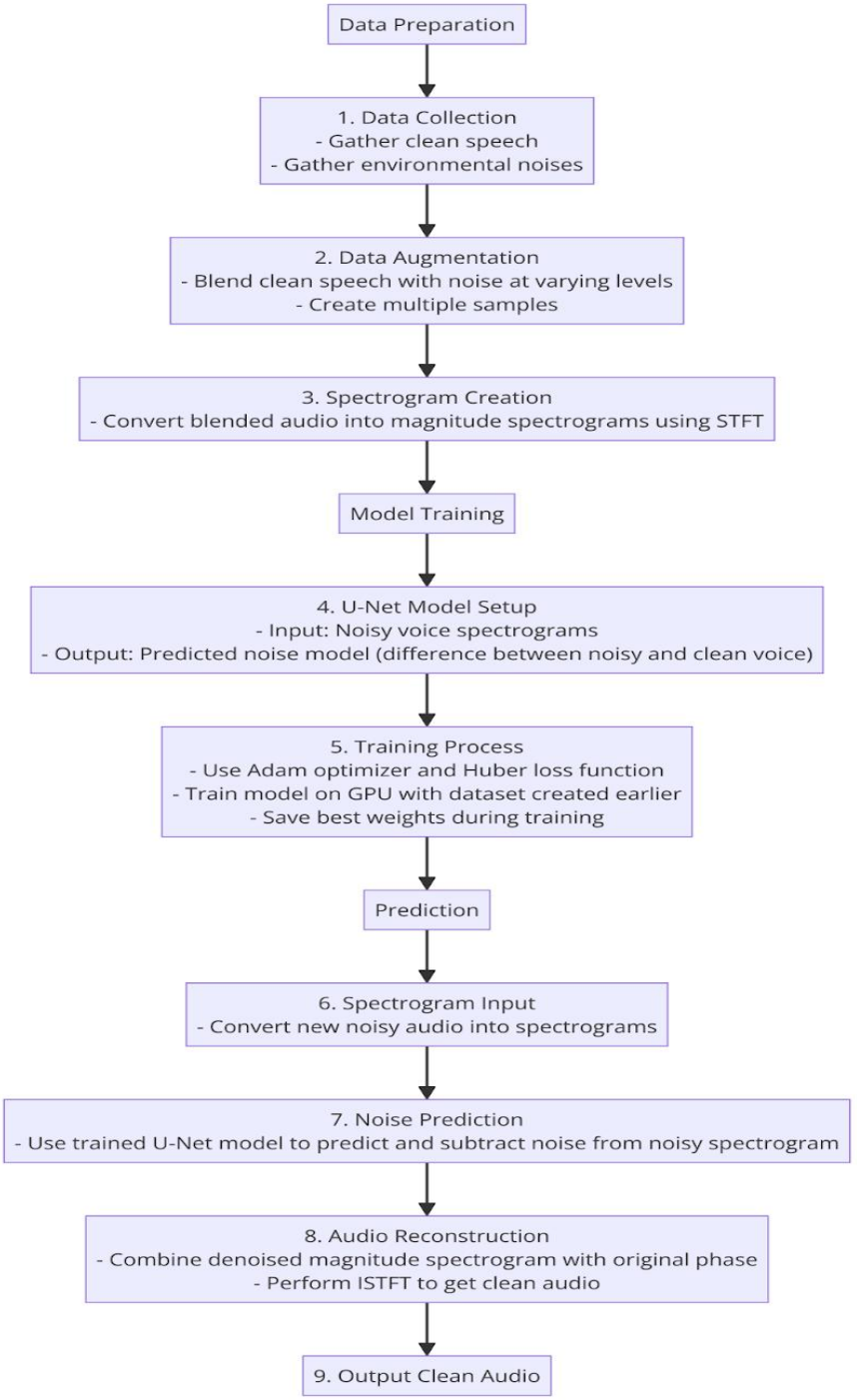

Fig.3.2 Diagram of Model workflow

3.3 Short-Time Fourier Transform (STFT)

Short-Time Fourier Transform (STFT): This technique divides the signal into short overlapping segments and applies the Fourier transform to each segment. The result is a spectrogram, a two-dimensional representation with time on one axis, frequency on the other, and intensity (amplitude) represented by color or grayscale.

What is a Spectrogram?

To process and improve audio, we need a way to represent sound visually. One effective method is using spectrograms. Think of a spectrogram as a picture of sound over time

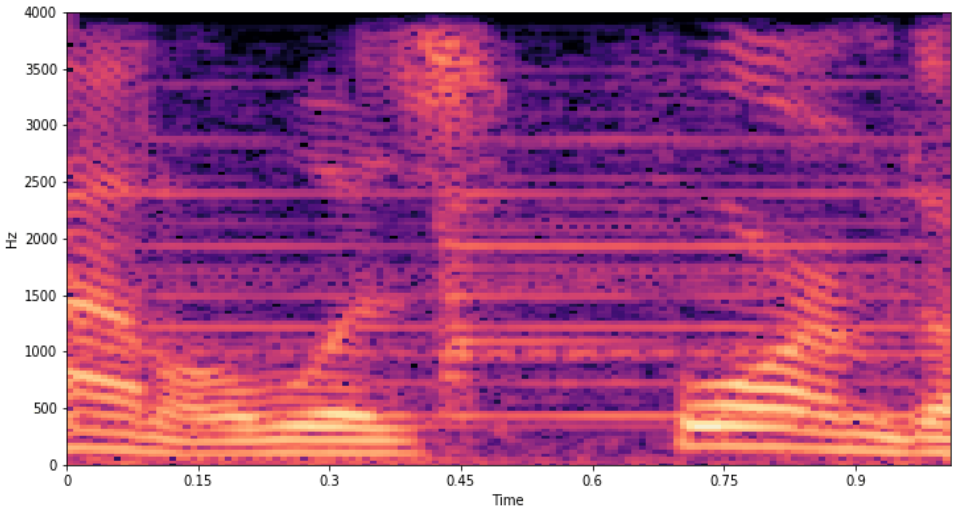

Fig.3.3 Diagram of Spectrogram

Axes: In a spectrogram, the horizontal axis represents time, and the vertical axis represents frequency (pitch).

Brightness: The brightness or intensity at any point on the spectrogram shows the strength (volume) of a particular frequency at a particular time.

3.4 Working Procedure

Data Creation Process:

1. **Load Audio Data**
   - Convert noise and voice audio files into numpy arrays for processing.
2. **Blend Clean Voice and Noise**
   - Randomly blend clean voice samples with noise to create noisy audio data.
3. **Save Blended Audio**
   - Save noisy voice, clean voice, and noise as `.wav` files for further use.
4. **Generate Spectrograms**
   - Convert audio signals into magnitude and phase spectrograms using STFT.
5. **Save Time-Series and Spectrograms**
   - Save time-series data and spectrograms as `.npy` files for future model training.

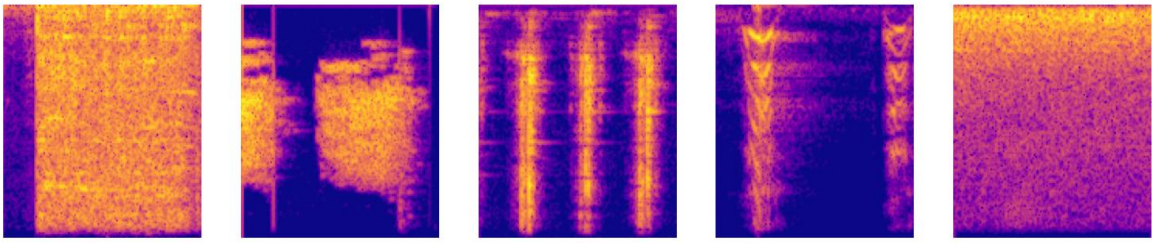

Fig.3.4 Diagram of Spectrogram Generated

Noisy Spectrogram = Noisy Voice Spectrogram – Clean Spectrogram

Training Steps:

1. Load Spectrogram Data

- Load noisy and clean voice spectrograms from `.npy` files.

2. Data Scaling and Reshaping

- Scale spectrograms between -1 and 1, and reshape for model input.

3. Train-Test Split

- Split the dataset into training and validation sets.

4. Model Initialization

- Initialize U-Net model (train from scratch or load pre-trained weights).

5. Model Training & Checkpointing

- Train the model, saving the best model based on validation loss.

6. Plot Loss

- Plot training and validation loss to track performance.

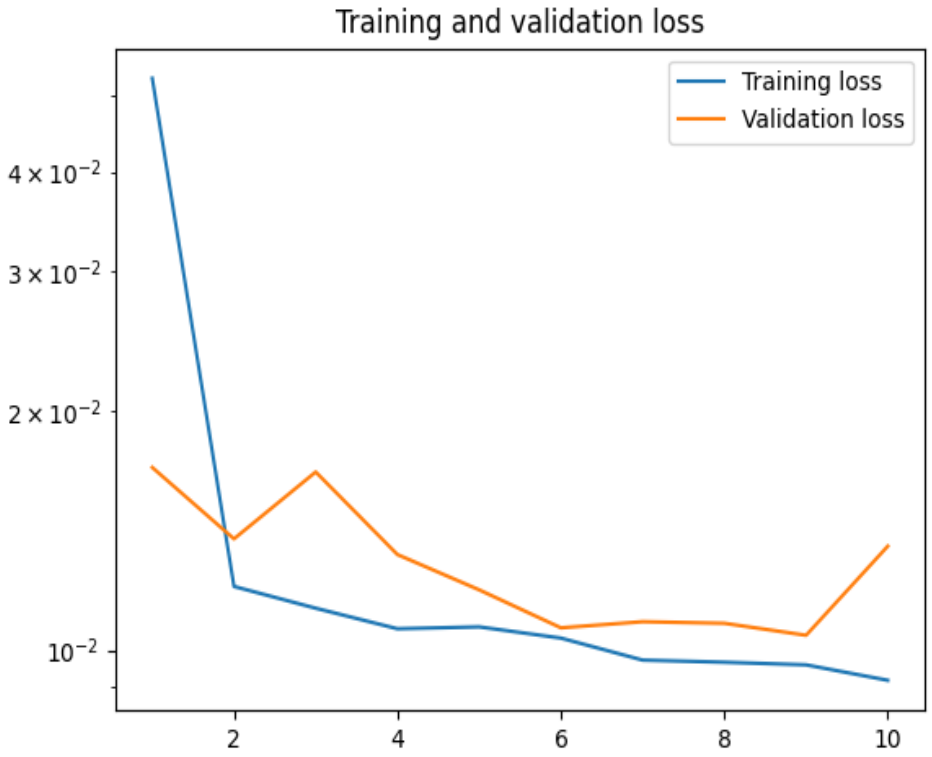

Fig 3.5 Training and Validation loss

Prediction:

- **Model Loading**: The function first loads the pre-trained model using the specified weights path.

- **Audio Preprocessing**: It converts the noisy input audio files to a numpy array, then calculates their magnitude and phase spectrograms.

- **Denoising**: The magnitude spectrogram is scaled globally to normalize it, then reshaped to fit the model's input structure. The model predicts the noise model, which is rescaled and subtracted from the noisy spectrogram to isolate the denoised voice spectrogram.

- **Audio Reconstruction**: Using the denoised magnitude spectrogram and the original phase, the function reconstructs the denoised audio through inverse transformation.

- **Saving Output**: Finally, the denoised audio is saved to the specified output path in WAV format

Chapter 4

Results and Conclusion

3) NB_samples = 512 Sample rate = 8000 kHz min_duration = 1.0 frame_length = 8064

hop_length_frame = 8064 hop_length_frame_noise = 4000 n_fft = 255 n_fft = 255 hop_length_fft = 63

Epoch	Training Loss	Validation Loss	Mean Absolute Error	Mean Squared Error
20	0.012259	0.019446	0.154298	0.038898

Metric's	snr_mean SNRovl	segsnr_mean SNRseg
Actual	7.0367	7.3021
Predicted	-2.0601	-3.7439

Metric's	ref_wav	deg_wav
Actual	2.3312	1.9416
Predicted	1.3467	1.2634

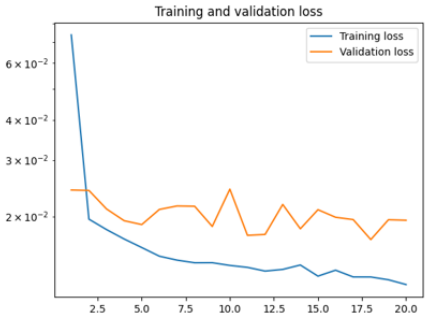

Fig.4.1 Snapshot of Result

2) NB_samples = 512 Sample rate = 8000 kHz min_duration = 1.0 frame_length = 8064

hop_length_frame = 8064 hop_length_frame_noise = 4000 n_fft = 255 n_fft = 255 hop_length_fft = 63

Epoch	Training Loss	Validation Loss	Mean Absolute Error	Mean Squared Error
20	0.012416	0.038669	0.223428	0.07737

Metric's	snr_mean SNRovl	segsnr_mean SNRseg
Actual	7.9516	8.3136
Predicted	-0.0064	-5.4073

Metric's	ref_wav	deg_wav
Actual	2.3684	1.9823
Predicted	1.3404	1.2611

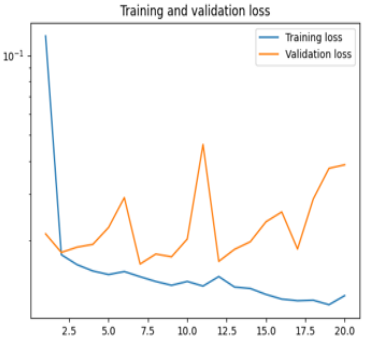

Fig.4.2 Snapshot of Result

4) NB_samples = 1024 Sample rate = 8000 kHz min_duration = 1.0 frame_length = 8064

hop_length_frame = 8064 hop_length_frame_noise = 4000 n_fft = 255 n_fft = 255 hop_length_fft = 63

Epoch	Training Loss	Validation Loss	Mean Absolute Error	Mean Squared Error
20	0.010449	0.015202	0.131656	0.030405

Metric's	snr_mean SNRovl	segsnr_mean SNRseg
Actual	8.2469	8.3397
Predicted	-0.7762	-4.2692

Metric's	ref_wav	deg_wav
Actual	2.4097	2.0287
Predicted	1.4395	1.2998

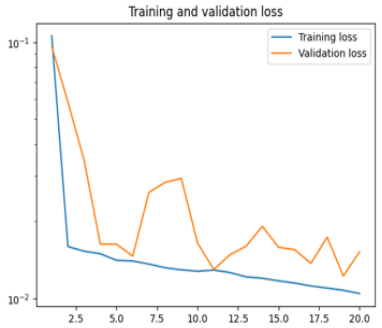

Fig.4.3 Snapshot of Result

5) NB_samples = 2048 Sample rate = 8000 kHz min_duration = 1.0 frame_length = 8064

hop_length_frame = 8064 hop_length_frame_noise = 4000 n_fft = 255 n_fft = 255 hop_length_fft = 63

Epoch	Training Loss	Validation Loss	Mean Absolute Error	Mean Squared Error
20	0.007596	0.013376	0.114674	0.02677

Metric's	snr_mean SNRovl	segsnr_mean SNRseg
Actual	7.2251	7.5023
Predicted	-6.5708	-5.8496

Metric's	ref_wav	deg_wav
Actual	2.3604	1.9734
Predicted	2.7988	2.5300

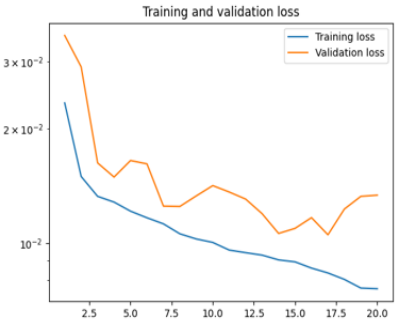

Fig.4.4 Snapshot of Result

6) NB_samples = 4096 Sample rate = 8000 kHz min_duration = 1.0 frame_length = 8064

hop_length_frame = 8064 hop_length_frame_noise = 4000 n_fft = 255 n_fft = 255 hop_length_fft = 63

Epoch	Training Loss	Validation Loss	Mean Absolute Error	Mean Squared Error
20	0.005993	0.00873	0.086696	0.017459

Metric's	snr_mean SNRovl	segsnr_mean SNRseg
Actual	7.7063	7.8116
Predicted	-1.5115	-4.0320

Metric's	ref_wav	deg_wav
Actual	2.3840	1.9997
Predicted	1.6818	1.4174

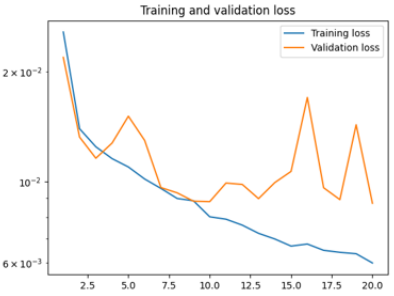

Fig.4.5 Snapshot of Result

References

[1] K. J. Piczak, "ESC: Dataset for Environmental Sound Classification," in *Proc. ACM Int. Conf. on Multimedia*, Orlando, FL, USA, Nov. 2015, pp. 1015-1018.

[2] O.Ronneberger, P. Fischer, and T. Brox, "U-Net: Convolutional Networks for Biomedical Image Segmentation," in *Proc. Int. Conf. on Medical Image Computing and Computer-Assisted Intervention (MICCAI)*, Munich, Germany, Oct. 2015, pp. 234-241.

[3] Jansson, E. Humphrey, N. Montecchio, R. Bittner, A. Kumar, and T. Weyde, "Singing voice separation with deep U-Net convolutional networks," in *Proc. Int. Conf. on Music Information Retrieval (ISMIR)*, Suzhou, China, Oct. 2017, pp. 745-751.

[4] N. Siddique, S. Paheding, C. P. Elkin, and V. Devabhaktuni, "U-Net and its variants for medical image segmentation: A review of theory and applications," *IEEE Access*, vol. 9, pp. 82031-82057, 2021.

[5] Y. Deng, Y. Hou, J. Yan, and D. Zeng, "ELU-Net: An efficient and lightweight U-Net for medical image segmentation," *IEEE Access*, vol. 8, pp. 123045-123053, 2020.