Speech Dereverberation via Generative Adversarial Training

1st Meet Sable
Student
DAIICT
Gandhinagar, India
201901442@daiict.ac.in

2nd Professor Hemant Patil
Mentor and Guide
DAIICT
Gandhinagar, India
hemant_patil@daiict.ac.in

I. Motivation

In speech communication, many problems are tough to solve with only mathematics. If enough data is available some deep learning framework can be used to solve the problem. Now, the problem arises when the data isn't consistent. In case of audio, the recording environment is also as crucial as the source of the sound. For the creation of database not every sound can be recorded in ideal condition, where only the source's sound is present and there is no reverb from the surrounding walls. These variation in sound recording sometimes might result in bad performance of the model. [1] So the main motivation behind this project is to build and train a GAN to remove the reverb from the Audio. [2]

II. Mathematics

Let, the source signal be x(t), impulse response of the environment be h(t) and the recorded audio be y(t). When the signal travels through an environment it convolves with the impulse response of that environment, so we can mathematically write eq. (1).

$$y(t) = x(t) * h(t) \tag{1}$$

Our objective is to remove the h(t) from the signal. We know that in the Fourier domain convolution becomes multiplication, which will be easier to remove.

$$\begin{aligned} FT\{y(t)\} &= FT\{x(t) * h(t)\} \\ FT\{y(t)\} &= FT\{x(t)\} \times FT\{h(t)\} \\ Y(f) &= X(f) \times H(f) \end{aligned} \tag{2}$$

Now, if we look at the eq. (2), it is just the multiplication between Fourier transform of the clean audio and the impulse response. In order to remove the impulse response we can multiply the eq. (2) with inverse of impulse response on both sides,

$$\begin{aligned} Y(f) \times H(f)^{-1} &= X(f) \times H(f) \times H(f)^{-1} \\ X(f) &= Y(f) \times H(f)^{-1} \end{aligned} \tag{3}$$

Based on the eq. (3), our main aim is to find out the inverse of the room impulse response from the provided audio with reverb. For this task, I have used a GAN, that will generate the inverse impulse response from the input reverberated audio.

III. Dataset

For the training of the GAN, two datasets were chosen:

- LibriSpeech ASR corpus:
  This dataset consist of 1000 hours of english speech. Dataset is divided into different folders based on the use i.e. dev-clean, dev-other, test-clean, test-other, train-clean-100, train-clean-360, train-other-500. For this project, only dev-clean folder is considered, it has 2703 different english speech filed ranging from length of 3 - 10 seconds per file. The sampling rate of all the audio files is 16,000Hz.
- Room Impulse Response and Noise Database:
  This dataset consist of simulated and real room impulse responses, isotropic and point-source noises. For the purpose of this project, I have taken the simulated impulse responses from this dataset. There are 60,000 simulated impulse responses representing different sizes and positioning of the mic and source in the room. The sampling rate of the impulse response is also 16,000Hz.

For this project, randomly a speech audio and an impulse response is chosen, and the convolution of that is fed into the gan. Both of this datasets are available freely available on openslr.org.

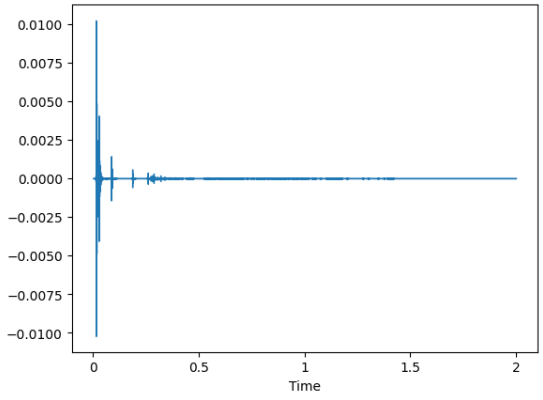
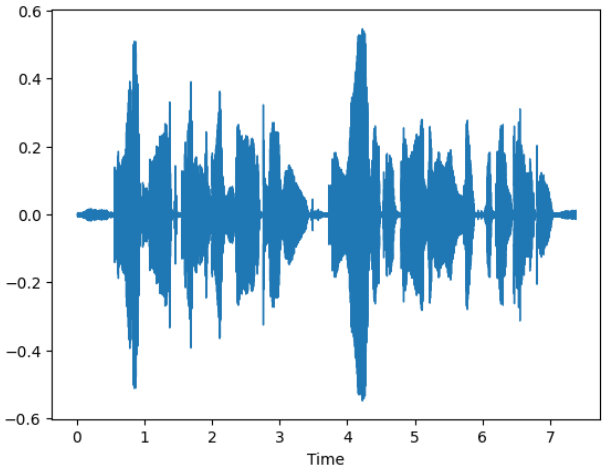
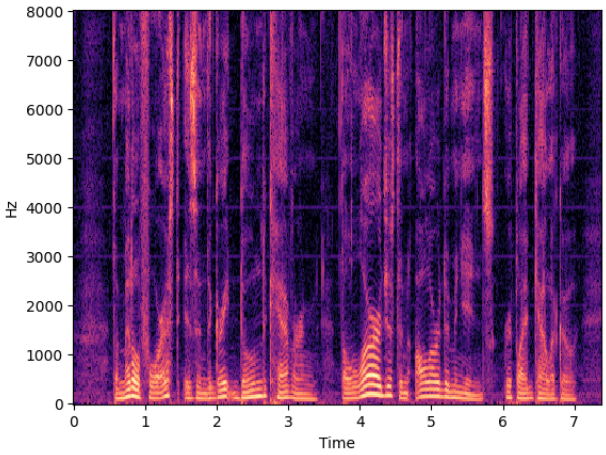


Fig. 1: Sample Impulse Response

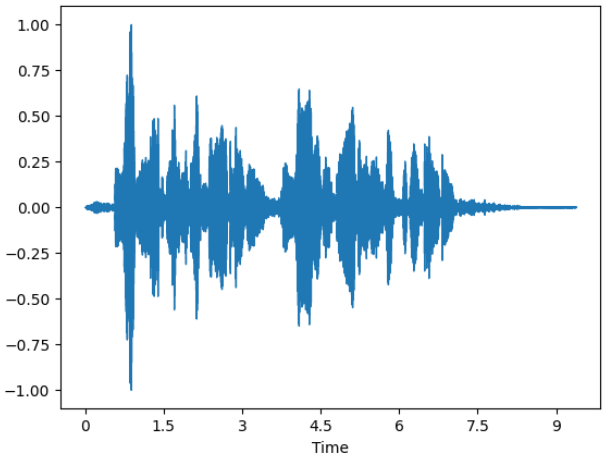


(a) Waveform

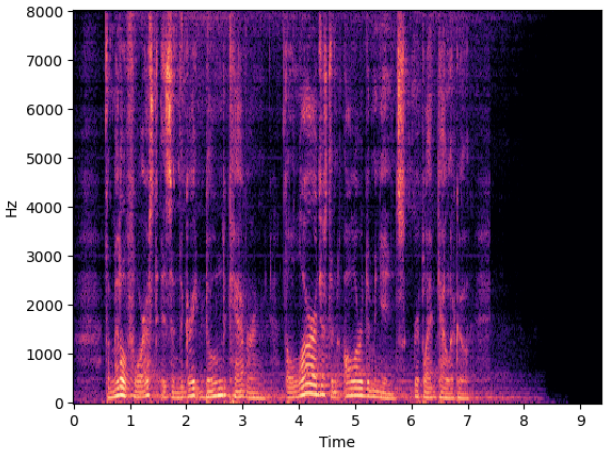


(b) STFT (window length = 35ms, hop length = 16ms)

Fig. 2: Clean speech



(a) Waveform



(b) STFT (window length = 35ms, hop length = 16ms)

Fig. 3: Reverberated speech

IV. Experimental Setup

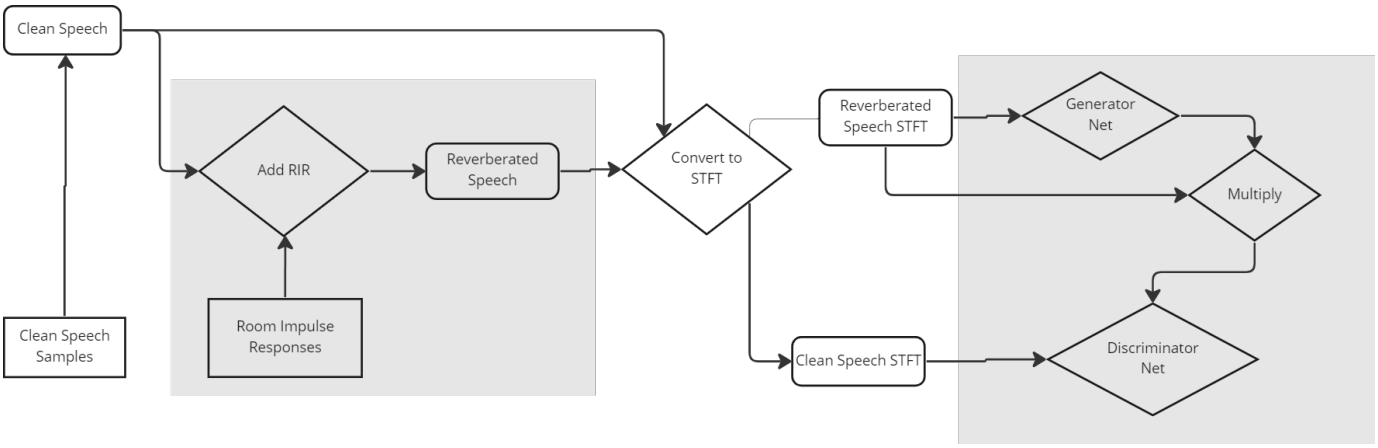
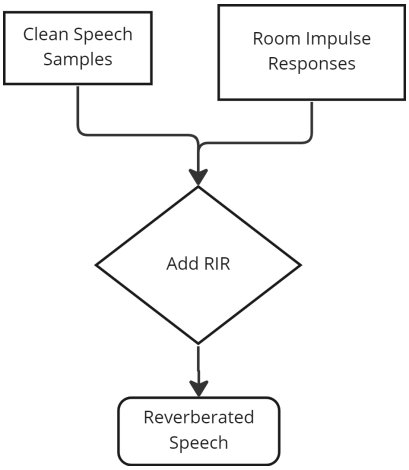


Fig. 4: Overall structure of the model

Intially we have 2 datasets containing audio files for clean speech and room impulse responses. From the datasets random impulse response and clean speech is chosen.

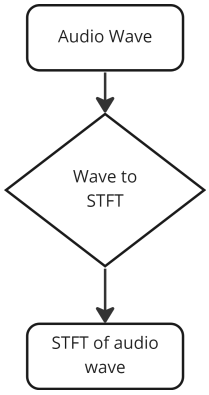
A. Add Reverb



The reverberated audio is obtained by doing convolution of impulse response and clean speech.

B. Conversion to STFT

Audio waveform is converted to Short time Fourier Transform. For the tranform, window length of 35ms, hop length of 16ms and resolution NFFT = 512 is taken. Example of resultant spectogram is shown in fig. [2b] & [6b].



C. Generator Network

Generator comprises of 5 convolutional layers, 2 stacked Bi-LSTM and one fully connected layer as shown in the fig. [5]. STFT magnitude spectrum of the audio wave of dimension 257 x length (will be dependent on the duration of speech wave).

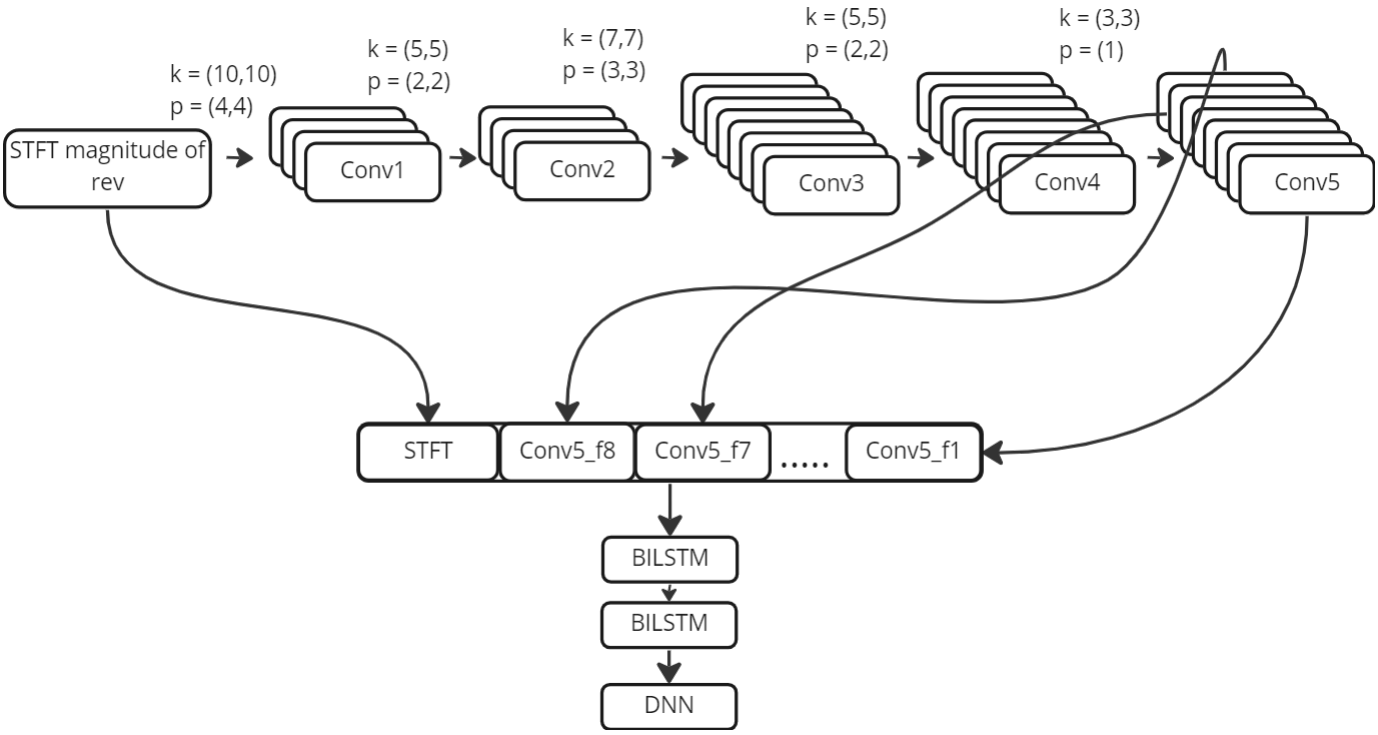
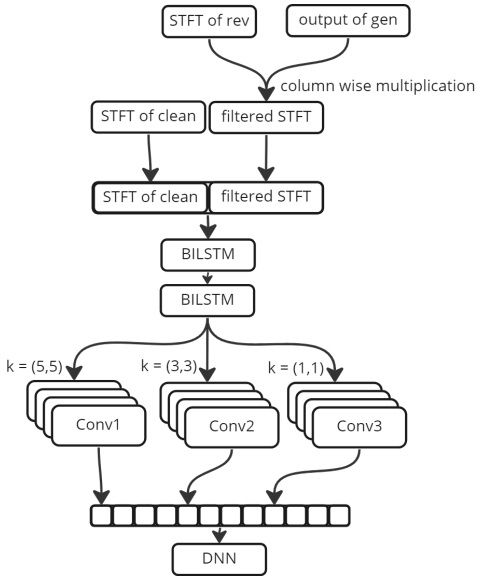


Fig. 5: Generator net

Spectrum passes through 5 convolutional layers of kernal sizes (10,10), (5,5), (7,7), (5,5) and (3,3) in order and the padding is chosen [(4,4), (2,2), (3,3), (2,2) and (1,1)] such that the height remains the same as input spectrum. Output of the convolution is then concatenated on the time axis and the STFT magnitude spectrum is concatenated at the front. This 2d tensor is then passed into the 2-stacked Bi-LSTM. Output of the Bi-LSTM will be a 2d tensor of dimension 4x257 (hidden layer). This tensor is flattened and then passed into the fully connected layer and output of the that will be a vector of length 257. Which supposedly represents inverse of room impulse response in Fourier domain.

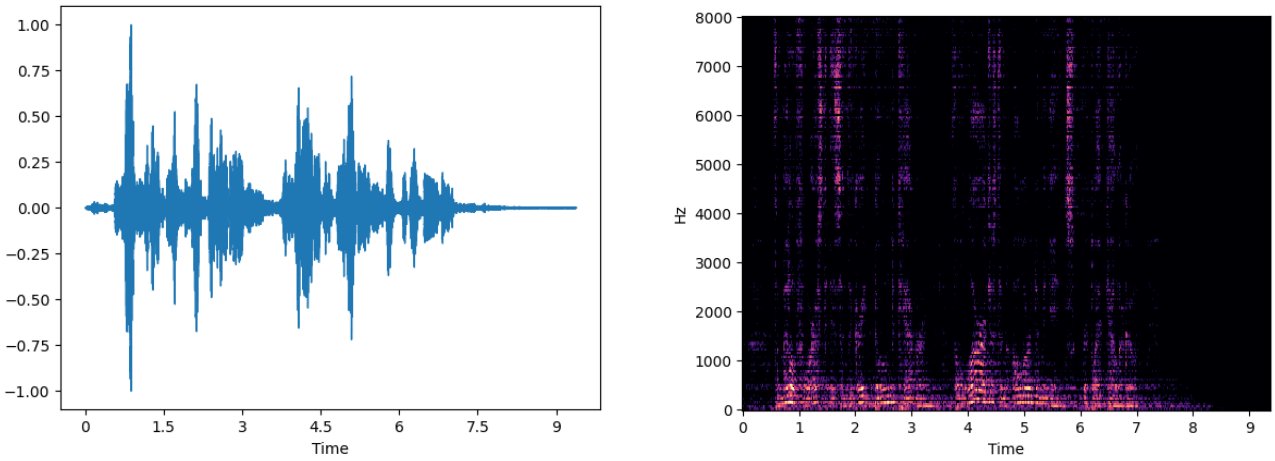
D. Discriminator



Discriminator network comprises of 1 2-stacked Bi-LSTM, 3 convolutional layers and 1 fully connected layer as shown in the figure [IV-D]. The output from the generator is multiplied with the STFT spectrum of the reverberated audio and we get filtered STFT. This filtered STFT and the clean audio STFT is passed to the discriminator network. In discriminator, clean and filtered spectrum are concatenated over the time axis. Then we pass this tensor through the 2-stacked Bi-LSTM. In this case, we will take the output instead of the hidden layer from the Bi-LSTM, which will be a tensor of 257xlen (dependent on the duration of the audio).Then we will process this tensor simultaneously via 3 convolutions layers of kernel sizes (5,5), (3,3) and (1,1) respectively. Each convolutioanl layer has 4 out channels. So now we will have 4(channels) $\times$ 3(layers) = 12 2d tensors. We will apply global max polling over all the 12 tensors, then these 12 values are passed into a fully connected layer with 1 output node, which will range from 0 to 1, where output closer to 0 means fake, and vice-versa.

V. Result

The GAN was trained over 25,000 iterations, which took around 8 hours. Result obtained was as following, Results are in this drive-folder : https://drive.google.com/drive/folders/1bBEFAWWtEJg4e_4zC55YMrEPoU3wfFlh?usp=share_link



(a) Waveform

(b) STFT (window length = 35ms, hop length = 16ms)

Fig. 6: Filtered speech waveform

VI. Reproducibility

The codes for the reproducability of the results are uploaded on kaggle:

- Helper functions : https://www.kaggle.com/code/meetsable/speech-gan-helper-funcs
  Contains functions for tasks such as converting to stft, dividing into frames and other miscellaneous functions
- GAN classes : https://www.kaggle.com/code/meetsable/dereverberator-gan
  Contains the generator, discriminator and training loop definitions
- train notebook : https://www.kaggle.com/code/meetsable/dereverberation-with-gat
  Notebook where the training was done, also has the models for 10,000 and 25,000 iterations saved in kaggle workspace

References

[1] M. J. Bianco, P. Gerstoft, J. Traer, E. Ozanich, M. A. Roch, S. Gannot, and C.-A. Deledalle. Machine learning in acoustics: Theory and applications. *The Journal of the Acoustical Society of America*, 146(5):3590–3628, 2019. Accessed: 2022-11-26.

[2] C. Li, T. Wang, S. Xu, and B. Xu. Single-channel speech dereverberation via generative adversarial training. *CoRR*, abs/1806.09325, 2018. Accessed: 2022-11-26.