A Study on Effectiveness of Deep Neural Networks for Speech Signal Enhancement in Comparison with Wiener Filtering Technique

1st Vijaya Kumar Padarti

Assistant Professor

Dept of Electronics & Communications

V R Siddhartha Engineering College

Andhra Pradesh, India

2nd Gnana Sai Polavarapu

Dept of Electronics & Communications

V R Siddhartha Engineering College

Andhra Pradesh, India

3rd Madhurima Madiraju

Dept of Electronics & Communications

V R Siddhartha Engineering College

Andhra Pradesh, India

4th Veera Venkata Naga Sai Nuthalapati

Dept of Electronics & Communications

V R Siddhartha Engineering College

Andhra Pradesh, India

5th Vinay Babu Thota

Dept of Electronics & Communications

V R Siddhartha Engineering College

Andhra Pradesh, India

6th Subramanyam Veeravalli

Dept of Electronics & Communications

V R Siddhartha Engineering College

Andhra Pradesh, India

Abstract: This paper intends to provide the optimum method between the Wiener filtering method and the Neural networks method for speech signal enhancement. A speech signal is highly susceptible to various noises. Many denoising methods include removal of high frequency components from the original signal. But this leads to removal of few parts of original signal. Thus, the quality of the signal reduces which is highly undesirable. Our main objective is to denoise the signal while we enhance its quality. Two methods, namely fully connected and convolutional neural network methods are compared with the Wiener filtering method and the most suitable technique will be suggested. To compare the output signal quality, we compute Signal to Noise ratio (SNR) and Peak Signal to Noise Ratio (PSNR). An advanced version of MATLAB with the advanced toolboxes such as Deep Learning toolbox, Audio toolbox, Signal Processing toolbox etc will be utilized for speech denoising and its quality enhancement.

Keywords – Wiener filtering technique, Neural networks, MATLAB, PSNR, SNR, Signal Processing, Audio Toolbox.

I. INTRODUCTION

The 5 fundamental senses i.e., sense to hear, sense to see, sense to smell, sense to taste and sense to touch, perceive the information from the environment and human brain processes this information to create a precise response.

Sound acts as an information provider to these senses. The information that is transmitted has to be free of noises to get a better understanding of the external environment. Noise can be described as any unwanted information which hinders the ability of the human body to process the valuable sensory information. Hence an uncorrupted sound becomes essential for proper interaction of humans with their external world. The primary focus is on speech signals which are information providers in various communication systems. During the transfer of signals, distortion by some unwanted signals causes loss of useful data and information stored in the signals. There are many real time noise signals such as the noise of a mixer grinder, washing machine and vehicles which have to be reduced to retrieve the wanted information. The frequency of speech signals ranges from 85Hz to 255 Hz. Typical male voice ranges in between 85-180 Hertz whereas the female voice ranges in between 165-255Hz. Babies have even higher ranges of frequency reaching up to 1000Hz in a few cases. [1]

Speech denoising refers to the removal of background content from speech signals. The goal of speech denoising is to produce noise-free speech signals from noisy recordings, while improving the perceived quality of the speech component and increasing its intelligibility [2]. Speech denoising can be utilized in various applications where we experience the presence of background noise in communications. For example, Hearing aids,

Telecommunications, Speech recognition applications etc [3].

A number of techniques have been proposed based on different assumptions on the signal and noise characteristics in the past, but in this paper, we shall compare 2 main methods, Wiener filtering technique, and Neural networks method. For neural networks technique we will consider 2 types of networks, Fully connected network and convolutional neural network. We compute PSNR and SNR values for these 3 techniques to compare the denoised signal quality.

II. BACKGROUND

A. Wiener filtering method

One of the notable techniques of filtering that is widely used in signal enhancement methods is Wiener Filtering. The key principle of Wiener filtering, essentially, is to take a noisy signal and acquire an estimate of clean signal from it. The approximate clean signal is acquired by reducing the Mean Square Error (MSE) between the estimated signal and desired clean signal [4].

The transfer function obtained in frequency domain is

$$H(\omega) = \frac{P_s(\omega)}{P_s(\omega) + P_v(\omega)} \quad (1)$$

Where $P_s(\omega)$ represents the Power spectral density of clean signal, while $P_v(\omega)$ represents the Power Spectral density of noise signal. Here, the signal “s” and noise “v” are considered to be uncorrelated and stationary.

The Signal to Noise Ratio (SNR), which is used to detect the quality of a signal, is defined as

$$SNR = \frac{P_s(\omega)}{P_v(\omega)} \quad (2)$$

Substituting SNR in the above transfer function, we obtain,

$$H(\omega) = \left[1 + \frac{1}{SNR}\right]^{-1} \quad (3)$$

One of the popular applications of Wiener filtering technique is Global Positioning system (GPS) and Inertial Navigation System. Wiener filter, which is also used in geodesy to denoise gravity records, is used in GPS to model only those time variabilities that are significant when adapted to noise level of data [5].

Signal coding applications is one field Wiener filter is widely used in. In signal processing and broadly engineering applications too, Wiener filter is considered to be a great tool for speech applications due to its accurate estimation characteristic. This filter can further be adapted to serve

different purposes like satellite telephone communication [6].

If we dive into the world of electronics and communication more, the Wiener filter has a range of applications in signal processing, image processing, digital communication etc like System identification, Deconvolution, Noise reduction and Signal detection [7].

Specifically in image processing, Wiener filter is a quite popular technique used for deblurring, attributed to its least-mean-squares technique. The blurriness in images that is caused as a result of motion or unfocused lens is removed using this filter. And since it returns mathematically and theoretically the best results, it also has applications in other engineering fields [8].

1) Algorithm

To denoise a speech signal using Wiener filtering technique, we first fetch a clean audio signal file and a noise signal file from the Audio datastore in MATLAB. We then extract a segment from the noisy signal and add it to the clean signal to make it a noisy speech signal which is given as input to the wiener filter. Wiener filter performs denoising of the speech signal and then we visualize the Output signal.

Fig 1 Wiener method Flow chart

In order to compute Peak SNR and SNR values, the output and input signals are given to PSNR function which is a built-in function in MATLAB. The PSNR function is as given by

$$[peakSNR, SNR] = psnr(output, input)$$

The Wiener filter has a tendency to inverse the filter at frequencies where SNR is high. Furthermore, the frequency response of the Wiener filter is such that, at frequencies where SNR is low, that is, noise power is more, the gain of the filter decreases, and the output is limited, causing noise-reduction. Correspondingly, for high SNR, that is, when signal power is more, the gain becomes nearly one (~1) and output sought is very close to input. Another drawback is that at all given frequencies, the Wiener filter requires a

fixed frequency response. One more shortcoming in the Wiener filter is that before filtering, the power spectral density of both clean and noise signals has to be estimated. Noise amplification is also a problem. [6,9-11]

B. Deep neural networks

Deep learning is a part of machine learning with an algorithm inspired by the structure and function of the brain, which is called an artificial neural network. Artificial neural networks are the statistical model inspired by the functioning of human brain cells called neurons. The very first research was published by Alexey Grigorevich Ivakhnenko while working on deep learning network in mid-1960's. Deep learning is suited over a range of fields such as computer vision, speech recognition, natural language processing, etc [12].

A neural network mimics the human brain and consists of artificial neurons, also known as nodes. These nodes are placed next to each other in three layers: The input layer, The hidden layer(s), and the output layer. There can be multiple hidden layers and it depends on the model. All the nodes are provided with information in the form of input. At each node, the inputs are multiplied with some random weights and are computed, then a bias value is added to it. Finally, activation functions such as ReLU function, are applied to determine which neuron to eliminate.

While deep learning algorithms feature self-learning representations, they depend upon neural networks that mimic the way the brain processes the information. During the training process, algorithms use random unknown elements in the input to extract features, segregate objects, and find useful data patterns. Much like training machines for self-learning, this occurs at multiple levels, using the algorithms to build the models. Deep learning models utilize several algorithms. Although none of the networks is considered perfect, some algorithms are preferred to perform specific tasks. Some commonly used Artificial neural networks are Feedforward Neural Network, Convolutional Neural Network, Recurrent Neural Network, Autoencoders [13].

There are also some disadvantages of deep learning. Very large amount of time is required to execute a deep learning model. Depending upon the complexity sometimes it may take several days to execute one model. Also, for small datasets the deep learning model doesn't suit.

There are various applications of deep learning such as Computer vision, Natural language processing and pattern recognition, Image recognition and processing, Machine translation, Sentiment analysis, Question Answering system, Object Classification and Detection, Automatic Handwriting Generation, Automatic Text Generation etc.

1) Algorithm

We first fetch clean and noisy audio files from the Audio Datastore in MATLAB, and then we extract a segment from the noisy audio and add it to the clean audio signal. This will be the input given to the Deep learning network.

Fig 2 Neural Networks Block Diagram

We utilize Short Time Fourier Transform (STFT) to transform the audio signals from time domain to frequency domain. The magnitudes are extracted and then fed to the neural network. Then the output signal which is the denoised and enhanced version of the input noisy signal is converted back into the time domain using the Inverse STFT.

An example speech signal is shown in the below figure 3. Clearly it can be seen that the amplitudes vary significantly with time, i.e., there will be huge variations frequently in the signals like music and speech. This is the reason we utilize Short Time Fourier transformation technique.

Fig 3 Sample audio signal

Deep learning network is modelled using two types of networks, fully connected and convolutional neural network.

For any model, the network first needs to be trained so that it learns its function to segregate the noise segments from the audio segments. For training the model, we consider a sample signal and then set the required parameters such as learning rate, number of epochs, batch size etc. Once the model completes its training, it has to be tested. For testing phase, we feed the model with another set of samples which were not given in the training phase and observe the outputs.

Fig 4 Neural Networks Flow chart

To compare the efficiency of the two models, we compute PSNR and SNR values using psnr function which is a built-in function in MATLAB.

$$[peakSNR, SNR] = psnr(output, input)$$

We also use another in-built function, sound(), to listen to the audio signals. Besides this we also represent the signals with timing plots and spectrogram.

2) Fully connected network

Fig 5 Fully Connected Network

A fully connected neural network consists of a series of fully connected layers that connect every neuron in one layer

to every neuron in the other layer. For any network there are three layers, Input, Hidden and Output layers. The information perceived from the data is passed as input to the model, then the model is trained using this data by multiplying random weights and adding bias [14].

We define the number of hidden layers in the model. For our model, we have two hidden layers with 1024 neurons each. The model is trained on the training dataset and is passed through three Epochs with batch size of 128. Each of the hidden layers are followed by ReLu layers and batch normalization layers.

A clean audio file fetched from the audio datastore is corrupted with a noisy segment extracted from the noise signal. These signals are plotted in the fig 5. Then these signals are passed to the network model and then model is trained. Training process involves model passing through the given dataset and learning the model function. We also define learning rate, which is defined as the rate at which the model learns its function.

Fig 6 Clean and Noisy Audio

3) Convolutional neural network

Convolutional neural networks can be differentiated from other neural networks by their superior performance with image, speech, or audio signal inputs. They have three main types of layers: Convolutional layer, Pooling layer and Fully-connected layer.

The first layer of a convolutional network is the convolutional layer. After convolutional layer, we can have additional convolutional layers or the pooling layers. But the fully connected layer will be the final layer. The CNN complexity increases with each layer. But the model gets more accurate outputs. So, there should be some optimality.

The convolutional layer is the core building block of a CNN. It is where the major part of the computation occurs. It requires a few components: input data, a filter, and a

feature map. Pooling layers, also known as down sampling, conducts dimensionality reduction, reducing the number of parameters in the input. It is similar to the convolutional layer in processing and filtering the input, but the difference is that this filter does not have any weights. Instead, the kernel applies an aggregation function to the values within the receptive field, populating the output array.

There are two main types of pooling: Max pooling which selects maximum value and Average pooling which computes average value of data. A lot of information is lost in the pooling layer. But it also has a number of advantages to the CNN. They help to reduce complexity, improve efficiency, and limit risk of overfitting.

Fig 7 Convolved layer

In the fully-connected layer, each node in the output layer connects directly to a node in the previous layer. This layer performs the task of classification based on the features extracted through the previous layers and their different filters. While convolutional and pooling layers tend to use ReLu functions, FC layers usually use a SoftMax activation function to classify inputs appropriately, producing a probability from 0 to 1 [15].

For our convoluted network model, we have employed total 16 layers. Similar to the fully connected network, convolutional layers are followed by ReLu and batch normalization layers.

III. RESULTS

A. Wiener filtering technique

The Wiener model is provided with a clean and a noisy signal, and it yielded the following output. We can clearly see that, few high frequency components in the original audio signal are lost when denoised. This leads to reduced

quality. Also, the model resulted in a PSNR 20.0589 of and SNR of 2.4825 which are considered to be low compared to standards.

Fig 8 Wiener method Results

Fig 9 Wiener method Spectrogram

B. Fully connected network

1) Training Stage

In the training stage, our model is provided with training dataset, and is made to learn its function. The training progress is visualized as in figure 12 where we can see the internal approximation process.

Fig 10 Fully connected network Training Progress

After the training is complete, the time and frequency plots are visualized as in figures 13 & 14. We can see that the denoised version is almost approximately equal to the original clean signal.

Fig 11 Fully connected network model Result plot in the Training Stage

2) Testing Stage

Once the model is trained, we test the model with a dataset which is not used in the training stage. When our fully connected model is tested, it resulted a PSNR of 23.5416 and an SNR of 6.5651, which are greater than those of the wiener method. But yet they are lower than the acceptable standard values. The output time and spectrogram plots are visualized as in figures 15 & 16. The original and enhanced versions are nearly equal but not exactly equal. But when compared to wiener filtering

technique, it can be seen that fully connected network method yielded better results.

Fig 12 Fully connected model Result Spectrogram in the training stage

Fig 13 Fully connected model Result plot in the testing stage

Fig 14 Fully connected model Result Spectrogram in the testing stage

C. Convolutional neural network

1) Training Stage

Similar to that of fully connected model, our convolutional model is first trained with a dataset. The training progress window is visualized in the figure 17. It should be noted that the training process took longer time compared to fully connected model due to the number of layers and the complexity of the layers. Once training is completed, the time and frequency plots are visualized as in figures 18 & 19.

Fig 15 Convolved model training progress

Fig 16 Convolved model Result plot in the training stage

Fig 17 Convolved model Spectrogram in the training stage

2) Testing Stage

After successfully training the model, we test the model with a testing dataset. Testing dataset is a dataset which is not provided in the training dataset. This helps in evaluation of the model.

In the testing stage, the time and frequency plots are visualized as in figures 20 & 21 where the original clean audio signal, corrupted noisy signal and the enhanced speech signals are plotted. It can be clearly seen that our model effectively denoises the noisy signal and enhances its quality. When compared to the previous technique, i.e., Wiener filtering technique, where we have removed all the high frequency components, we have removed only the noisy components and retained the original high frequency components in the output signal. So, this model is effective compared to the wiener method.

Fig 18 Convolved model Result plot in the testing stage

Fig 19 Convolved model Spectrogram in the testing stage

The convolved model resulted as SNR of 7.6137 and a PSNR of 26.4451. When compared to wiener and fully

connected models, these values are higher. However, they are still lower than the acceptable standard values.

IV. CONCLUSION

We have built three models to employ Wiener filtering technique and neural networks for speech enhancement. The results from models for different samples are noted in the tables 1,2 & 3. From the results obtained we can clearly see that convoluted network performs better when compared to the other two models. But it requires very large amount of time for training and computation. We know that the resources are very limited and expensive to process such models. Besides this, requiring a very large computational time is a big disadvantage when it comes to real time applications. Also, when the model requires such huge resources, the model must also be very efficient, but the results obtained from the convoluted network model are not much satisfactory. The signal needs to be much more enhanced keeping the required resources and time in view. So, we still need to optimize the model for better results.

Sample	PSNR	SNR
1	20.0589	2.4825
2	20.7663	3.1900
3	20.4732	2.8969
4	20.8320	3.2557
5	21.2536	3.6772
6	20.6340	3.0577
7	20.2921	2.7157
8	20.6636	3.0872

Table 1 Wiener filtering Technique

Sample	PSNR	SNR
1	23.5416	6.5651
2	24.1829	7.2064
3	24.0188	7.0423
4	26.5571	7.7258
5	23.7959	6.8194
6	23.8697	6.8932
7	23.9390	6.9625
8	23.8521	6.8756

Table 2 Fully Connected model

Sample	PSNR	SNR
1	26.4451	7.6137
2	25.9861	7.1547
3	26.1930	7.3617
4	26.3692	7.5378
5	26.3822	7.5508
6	25.9799	7.1485
7	26.2276	7.3962
8	26.5732	7.7461

Table 3 Convolved method

REFERENCES

- [1] G. K Rajini, V. Harikrishnan, Jasmin Pemeena Priyadarisini M, S. Balaji, "A Research on Different Filtering Techniques and Neural Networks Methods for Denoising Speech Signals", IJITEE, ISSN: 2278-3075, Volume-8, Issue- 9S2, July 2019.
- [2] Ding Liu, Paris Smaragdis, Minje Kim, "Experiments on Deep Learning for Speech Denoising", INTERSPEECH, 2014
- [3] Francis G. Germain, Qifeng Chen, and Vladlen Koltun, "Speech Denoising with Deep Feature Losses", arXiv:1806.10522v2 [eess.AS], 14 Sep 2018.
- [4] M. A. Abd El-Fattah, M. I. Dessouky, S. M. Diab and F. E. Abd El-samie, "SPEECH ENHANCEMENT USING AN ADAPTIVE WIENER FILTERING APPROACH", Progress in Electromagnetics Research M, Vol. 4, 167–184, 2008.
- [5] Klos, A., Bos, M.S., Fernandes, R.M.S. *et al.* Noise-Dependent Adaption of the Wiener Filter for the GPS Position Time Series. *Math Geosci* **51**, 53–73 (2019). Available at <https://doi.org/10.1007/s11004-018-9760-z>.
- [6] S. China Venkateswarlu, Dr. K. Satya Prasad, Dr. A. SubbaRami Reddy, "Improve Speech Enhancement Using Weiner Filtering", Global Journal of Computer Science and Technology Volume 11 Issue 7 Version 1.0 May 2011.
- [7] This is from an online internet source which is available at https://en.wikipedia.org/wiki/Wiener_filter#Applications.
- [8] This is from an online internet source which is available at <https://www.clear.rice.edu/elec431/projects95/lords/wiener.html>.
- [9] Chavan, Karishma & Gawande, Ujwalla. (2015), "Speech recognition in noisy environment, issues and challenges: A review", 1-5. 10.1109/ICSNS.2015.7292420.
- [10] Article from an online internet source available at <https://www.owl.net.rice.edu/~elec539/Projects99/BACH/proj2/wiener.html>.
- [11] Anju Dubey and Michel Galley, "Experiments with Speech Enhancement Techniques", available at <https://www.ee.columbia.edu/~dpwe/classes/e4810-2002-09/projects/mg2016/report.html>.
- [12] This is from an online internet source which is available at <https://www.investopedia.com/terms/d/deep-learning.asp#:~:text=Deep%20learning%20is%20a%20subset,learning%20or%20deep%20neural%20network>.
- [13] This is from an online internet source which is available at <https://www.simplilearn.com/tutorials/deep-learning-tutorial/deep-learning-algorithm>.
- [14] This is from an online internet source which is available at <https://medium.com/swlh/fully-connected-vs-convolutional-neural-networks-813ca7bc6ee5#:~:text=%20Fully%20connected%20neural%20network%20%201%20A,such%20networks%20do%20tend%20to%20have...%20More%20>.
- [15] This is from an online internet source which is available at <https://www.ibm.com/cloud/learn/convolutional-neural-networks>