CHAPTER 3

TRADITIONAL ALGORITHMS FOR SPEECH ENHANCEMENT

3.1 SPECTRAL SUBTRACTION (SS)

The Spectral Subtraction (SS) method is a popular traditional technique for speech enhancement that decreases additive noise. This method subtracts the noisy signal spectrum from the average noise spectrum to remodel the power or magnitude spectrum, as shown in Figure 3.1. The clean speech signal spectrum is estimated by subtracting the noisy speech spectrum from the average noise spectrum. The noise spectrum, assumed to be relatively stable and slowly evolving, is estimated and updated during signal absence and noise presence periods.

Phase analysis of the noisy signal and the instantaneous magnitude spectrum is used for signal restoration in the time domain, translated via an inverse discrete Fourier transform (IDFT). Spectral subtraction may result in negative short-time power or magnitude spectra due to random noise changes; these negative values must be transformed into non-negative ones, distorting the recovered signal distribution. This distortion becomes more apparent as the signal-to-noise ratio decreases.

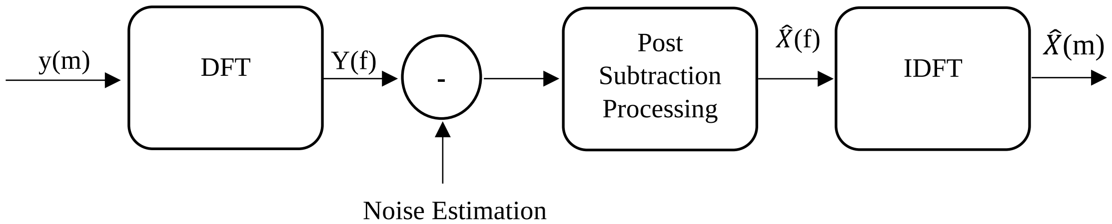

Figure 3.1 Block Diagram of Spectral Subtraction

Implementing SS involves a silence detector to detect signal inactivity. The Discrete Fourier Transform (DFT) converts the time domain to the frequency domain, followed by a magnitude operator. A Low Pass Filter (LPF) reduces distortion and noise variance before the IDFT transfers the processed signal back to the time domain, minimizing spectral subtraction distortions. The procedure for the SS method is shown in Figure 3.2.

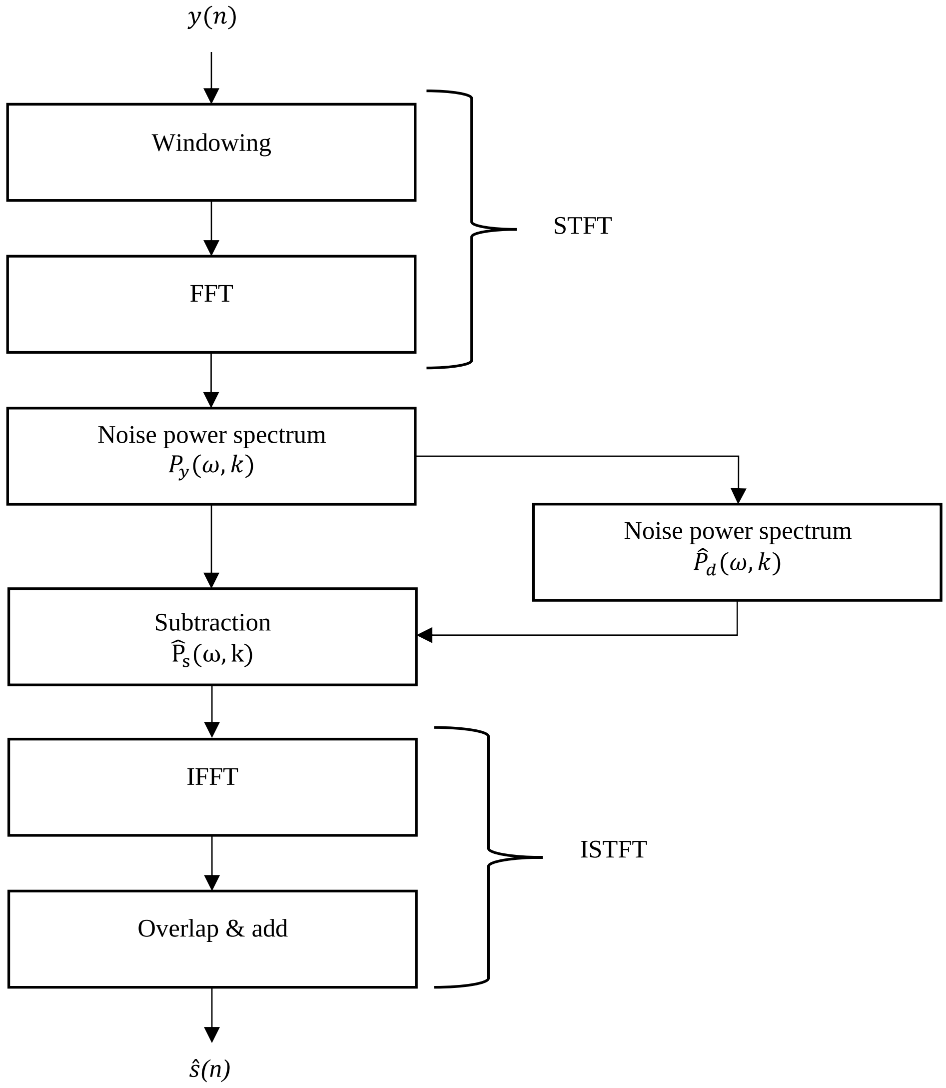

Figure 3.2. Flow Diagram of Spectral Subtraction Method

3.1.1 Principle of Spectral Subtraction Method

Noiseless speech consists of clean speech that has been distorted by an additive noise as given in equation 3.1

$$y(n) = x(n) + n(n) \tag{3.1}$$

The noisy speech sample, clean speech sample, and additive noise sample are represented by y[n], x[n], and n[n]. The additive noise is expected to have zero mean, and it is also unrelated to clear speech. As a result of its non-stationarity and time-variability, noisy speech signals are often processed frame-by-frame. A noisy signal in the STFT domain is represented by equation 3.2

$$Y(\omega, g) = X(\omega, g) + N(\omega, g) \tag{3.2}$$

Where,

ω - discrete frequency

g - frame number

Due to the simple nature of the problem, 'g' is ignored, assuming that the speech signal is segmented into frames.

When the received signal is eliminated from a noise estimate, the speech can be extracted and is given in equation 3.3

$$|\hat{X}(\omega)|^2 = |\hat{Y}(\omega)|^2 - |\hat{N}(\omega)|^2 \tag{3.3}$$

By modifying (3.3), equation of noisy speech spectrum and the Spectral Subtraction Filter (SSF) is given by equation 3.4

$$|\hat{X}(\omega)|^2 = 1 - \left(\frac{|\hat{N}(\omega)|^2}{|\hat{Y}(\omega)|^2}\right) |Y(\omega)|^2 = H^2(\omega)\, |Y(\omega)|^2 \tag{3.4}$$

where H(ω) varies in the range of $0 \leq H(\omega) \leq 1$ as given in equation 3.5

$$H(\omega) = \left\{max\ \left(0, 1 - \frac{|\hat{N}(\omega)|^2}{|\hat{Y}(\omega)|^2}\right)\right\}^{\frac{1}{2}} \tag{3.5}$$

The speech phase estimate is required to reconstruct the resultant signal. Given that the short-term phase is mainly ignored by human hearing, a popular phase estimation technique uses the noisy signal as the anticipated clean speech signal. Then, the frame of the speech signal is calculated as given in equation 3.6

$$\hat{X}(\boldsymbol{\omega}) = |\hat{X}(\boldsymbol{\omega})| e^{j<Y(\omega)} = H(\omega)Y(\omega) \tag{3.6}$$

While the Spectral Subtraction (SS) approach effectively reduces noise, it has significant drawbacks. Its performance heavily depends on accurate noise estimation, which often needs to be more attainable. The main issue with the SS approach is that the improved speech has a distracting audible tonal feature that interferes with human hearing and is called musical noise (Boll, 1979; Upadhyay & Karmakar, 2012). Sometimes, speaker identification technologies and human hearing find this musical noise more disturbing.

3.1.2 Multiband Spectral Subtraction Algorithm

The multiband approach applies spectral subtraction as a nonlinear technique, targeting specific noise-related frequency bands. Since noise impacts different frequencies unevenly, the speech spectrum is divided into N discrete bands, with spectral subtraction performed on each band separately. The speech signal is divided using appropriate windows in the time domain, and noise is subtracted from each frequency band. The modified bands are then recombined to produce the enhanced signal through the inverse FFT (IFFT) of the expanded spectrum. The clean speech signal is given by equation 3.7

$$|\widehat{X}_i(\omega_k)|^2 = |\widehat{Y}_i(\omega_k)|^2 - \alpha_i \, . \, \delta_i \, |\widehat{N}_i(\omega_k)|^2; \; a_i \le \omega_k \le e_i \quad (3.7)$$

Where,

ω_k - discrete frequencies

$|\widehat{X}_i(\omega_k)|$ - Clean speech power

$|\widehat{N}_i(\omega_k)|^2$ - estimated noise powers

a_i , e_i - initial and final frequency bins

α_i - subtraction factor

$\widehat{Y}_i(\omega_k)|$ - smoothed noisy speech spectrum

δ_i – additional subtraction band

Speech and musical signals are audio signals from non-stationary acoustic event sequences. Spectral subtraction produces time-varying musical tone noises as undesirable byproducts. Musical noise may be identified by the detection of fluctuations of signal in the time and frequency domains. The primary features of musical noise often consist of random bursts of sounds that are generally brief with tiny amplitudes.

The disadvantage of the spectral subtraction method includes fixed-value subtraction parameters that cannot be adjusted for changing noise characteristics and levels. Most of the noise introduced to speech has a spectrum that is not flat, making it challenging to optimize the settings. Multi-band spectral subtraction addresses this by adjusting the time and frequency domain parameters based on SNR. The performance of subtractive algorithms is evaluated using SNRs, PESQ scores, spectrograms, and informal listening tests. Literature indicates that conventional spectral subtraction leaves audible residual noise, reducing speech intelligibility and making it unsuitable for residual noise suppression.

3.2 MINIMUM MEAN SQUARE ERROR (MMSE)

According to Ephraim and Malah's (Malah 1985) proposal, the mean square error between the logarithms of the magnitude spectra of the actual and estimated signals should be minimized when estimating the clean signals, as shown in Figure 3.3. This approach reduces residual noise with minimal speech distortion, enhancing intelligibility using MMSE modulation magnitude estimation (MME) stimuli rather than Modulation Spectral Subtraction (ModSSub) stimuli. Unlike spectral subtraction-based methods, MME effectively reduces noise without introducing musical noise. The findings also show that ModSSub generates less musical noise than acoustic spectral subtraction by carefully selecting the modulation frame. Frame time is crucial in modulation frame-based approaches, with shorter durations generally improving quality. Studies demonstrate that the short-time modulation domain can effectively process speech instead of the short-time acoustic domain. MME provides efficient noise suppression without introducing distortion. Spectral subtraction, a well-researched technique for reducing additive noise, often introduces irritating artifacts known as musical noise. The MMSE short-time spectral amplitude estimator addresses this issue.

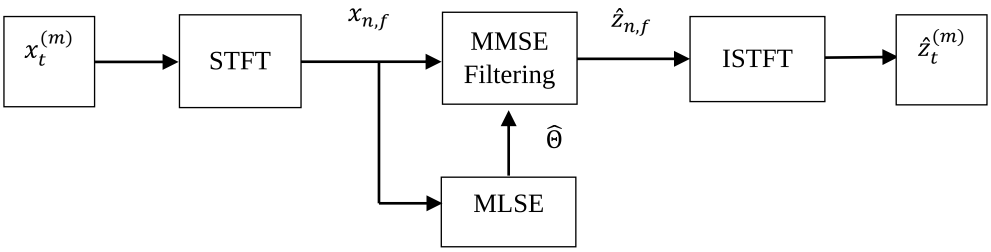

Figure 3.3 MMSE based Speech Enhancement

Consider the additive noise model in which clean speech is distorted by uncorrelated additive noise to generate noisy speech is given by equation 3.8

$$y(n) = x(n) + n(n) \quad (3.8)$$

Where $y(n)$, $x(n)$, and $n(n)$ are the noisy speech signal, clean speech, and noise signal, respectively, and 'n' represents a discrete-time index. Even though speech is non-stationary, it is considered to be quasi-stationary and allows the frame-wise processing of the noisy speech signal utilizing the ongoing STFT analysis provided by equation 3.9

$$Y(l,k) = \sum_{n=0}^{N-1} y(n + lZ)w(n)e^{-j2\pi nk/N} \quad (3.9)$$

Where

k - acoustic frequency index

l - acoustic frame index

N - Acoustic Frame Duration (AFD)

Z - Acoustic Frame Shift (AFS)

Speech processing typically uses AFS of 10-20 ms and AFD of 20-40 ms. The MMSE estimator effectively reduces background noise without adding musical noise, but its implementation suffers from suboptimal suppression factors, which can be improved for higher speech quality.

3.3 WIENER FILTER

The Wiener filter is an algorithm used in signal processing to approximate a target through Linear Time-Invariant (LTI) filtering of an observed noisy process under the assumption of known stationary signal, noise spectra, and additive noise. The mean square error between the intended and the estimated random processes is reduced using the Wiener filter.

In the Wiener filter, an unknown signal is statistically estimated from a related signal by filtering that known signal, relying on that estimated signal as an output, and then taking the estimate. For instance, the known signal can be a distorted unknown signal caused by additive noise. Wiener filters help estimate the underlying signal by removing noise from distorted signals. The Wiener filter employs a statistical-based method, where the solution is based on optimizing the mean-squared error to minimize the total error. The error is the difference between the reference signal and the filter output.

In actual speech-processing contexts, noise is an inherent component. The Wiener filter is the most fundamental single-channel speech enhancement technique (Jaiswal et al., 2022). It reduces the MSE between the estimated and original signals, requiring proper adjustment to handle speech misrepresentation. Noise reduction and speech enhancement techniques are essential to obtain a clear speech signal from corrupted ones.

In the statistical filtering problem, design a system with the output signal as close to the desired signal d(n). This is done by computing the estimation error e(n).

An FIR system is considered as shown in Equation (3.10)

$$\hat{d}(n) = \sum_{k=0}^{M-1} h_k y(n-k) \quad n = 0, 1, 2, \ldots \tag{3.10}$$

where

$\{h_k\}$ are the FIR filter coefficients

M is the number of coefficients

The filter coefficients $\{h_k\}$ is computed to minimize the estimation error, that is, $d(n) - \hat{d}(n)$, is minimized.

From Equation (1), the output signal $\hat{d}(n)$ can be obtained by convolving the impulse response of the system $\{h_k\}$ with the input signal y(n) as shown in equation (3.11)

$$\hat{d}(n) = h(n) * y(n) \tag{3.11}$$

where * denotes convolution.

The discrete-time Fourier Transform of h(n) and y(n) are represented as $H(\omega)$ and $Y(\omega)$ as shown in Equation (3.12)

$$\widehat{D}(\omega) = H(\omega)Y(\omega) \tag{3.12}$$

The estimation error at frequency ω_k is defined as shown in Equation (3.13)

$$\begin{aligned} E(\omega_k) &= D(\omega_k) - \widehat{D}(\omega_k) \\ &= D(\omega_k) - H(\omega_k)Y(\omega_k) \end{aligned} \tag{3.13}$$

The need to compute $H(\omega)$ arises as the mean-square error has to be minimized.

After computing the mean square error, then introducing the power spectrum of y(n) and cross power spectrum of y(n) & d(n), further simplifying the equation to find the optimal filter $H(\omega_k)$.

Wiener filter in the frequency domain is represented as shown in equation (3.14)

$$H(\omega_k) = \frac{P_{dy}(\omega_k)}{P_{yy}(\omega_k)} \tag{3.14}$$

Finally, after substituting the values of $P_{dy}(\omega_k)$ and $P_{yy}(\omega_k)$ in Equation (3.14), the Wiener filter in the frequency domain is obtained:

$$H(\omega_k) = \frac{P_{xx}(\omega_k)}{P_{xx}(\omega_k)+P_{nn}(\omega_k)} \tag{3.15}$$

The conventional Wiener filter's static assumptions may compromise noise reduction in dynamic environments, often resulting in oversmoothing and reduced speech clarity. To address this, the Iterative Wiener Filter iteratively refines estimates, enhancing adaptability and improving speech quality.

3.3.1 Iterative Wiener Filter (IWF)

The Iterative Wiener filter (Mouchtaris et al., 2004) is a speech enhancement algorithm that uses a recursive approach to estimate the speech signal from a noisy mixture. The IWF is an iterative algorithm applied repeatedly to the noisy mixture. The Wiener filter is updated with each iteration to estimate the speech signal better. This iterative approach allows the IWF to adapt to changes in the speech and noise signals. In the IWF method, the clean speech spectrum is estimated by

$$\hat{X}_{i+1(\omega_k)} = H_i(\omega_k)Y(\omega_k) \tag{3.16}$$

$H_i(\omega_k)$ – represents the Wiener filter obtained in i^{th} iteration.

The estimate of Wiener filter $H_i(\omega_k)$ is obtained based on the clean speech signal $\hat{x_i}(n)$. During the first iteration $\hat{x_i}(n)$ is initialized with the noisy speech signal y(n). The noisy speech signal y(n) is filtered by the newly obtained wiener filter $H_i(\omega_k)$ according to equation 3.16 to get the new enhanced speech signal $\widehat{x_{i+1}}(n)$. In the place of $\hat{x_i}(n)$, $\widehat{x_{i+1}}(n)$ is used, and the procedure is repeated until the noisy speech signal is enhanced. The stopping criterion is set to a predetermined maximum number of iterations to eliminate the convergence issues. The number of iterations is limited to 10. Once this limit is reached, the algorithm terminates whether it is converged or not.

3.4 TRADITIONAL HYBRID ALGORITHM

A hybrid model combining the Wiener filter, Wavelet Transform, and Least Mean Squares (LMS) algorithm (Ma & Nishihara, 2014; Venkateswarlu et al., 2021)is designed for speech enhancement to effectively reduce noise from a noisy speech signal while preserving the quality of the speech signal.

3.4.1 Wavelet Transform

The process begins with the wavelet transform. The Discrete Wavelet Transform (DWT) decomposes the noisy speech signal into multiple scales or levels. This decomposition separates the signal into different frequency components, with each scale capturing information at a specific frequency range. The wavelet transform decomposes the signal into different scales or levels using the DWT.

The noise is represented as n(n) and the clean speech signal is denoted as x(n). The noisy speech signal is $y(n) = x(n) + n(n)$

Let W_y represent the wavelet transform of the noisy speech signal $y(n)$, resulting in different scales $W_y(k, n)$ as shown in equation 3.17, where k represents the scale.

$$W_y(k,n) = DWT\ (y(n)) \tag{3.17}$$

3.4.2 Wiener Filter

A Wiener filter is applied for each scale obtained from the wavelet decomposition. The Wiener filter estimates the clean speech signal's power spectral density (PSD) and the noise PSD at each scale. It then applies a frequency-dependent gain to attenuate the noise while preserving the speech components. The Wiener filter adaptively adjusts the gains based on the estimated PSDs.

The Wiener filter estimates the gain to apply to each scale. Let $G(k, n)$ represent the gain at scale k and time index n for noise reduction.
$D(k,n)$ is the denoised signal obtained from the Wiener Filter. Applying the Wiener filter to each scale to obtain the denoised scales as given in equation 3.18

$$D(k,n) = G(k,n). W_y(k,n) \tag{3.18}$$

3.4.3 LMS Algorithm

The LMS algorithm further adapts and refines the Wiener filter's gains. The LMS algorithm continuously updates the filter coefficients based on the difference between the estimated clean speech and the noisy speech signal. This adaptive process helps the filter adapt to changing noise characteristics and improves noise reduction performance further.

The LMS algorithm updates the Wiener filter gains based on the estimated clean speech signal and the noisy speech signal at each scale. Let represent the error signal at scale and time index as shown in equation 3.19,

$$e(k,n) = x(n) - D(k,n) \tag{3.19}$$

The Wiener filter gains are adapted using the LMS update rule as shown in equation 3.20.

$$G(k,n+1) = G(k,n) + \mu. e(k,n). W_y(k,n) \tag{3.20}$$

where μ is the LMS adaptation step size.

3.4.4 Inverse Wavelet Transform

After filtering each scale with the Wiener filter and LMS algorithm, the filtered scales are combined using the Inverse Discrete Wavelet Transform (IDWT) to reconstruct the denoised speech signal. This reconstructed signal is expected to have reduced noise while preserving the essential speech information.

Finally, the denoised scales $D(k, n)$ are combined using the inverse discrete wavelet transform (IDWT) to reconstruct the denoised speech signal, as shown in equation 3.21.

$$s_{denoised}(n) = IDWT(D(k, n)) \tag{3.21}$$

This sequence describes a hybrid speech signal denoising model combining the Wiener filter, wavelet transform, and LMS algorithm. The wavelet transform separates the speech signal into different frequency components. The Wiener filter adapts gains based on estimated PSDs for noise reduction, while the LMS algorithm refines the process for dynamic noise. This hybrid approach effectively reduces noise across various types and intensities, preserving speech quality.

3.5 COMPARISON OF TRADITIONAL ALGORITHMS

Considering traditional techniques, the main issue with spectral subtraction is the distracting audible tonal feature in the enhanced speech. It uses fixed subtraction parameters, making it inflexible for changing noise levels. The MMSE filter introduces suboptimal suppression factors, while Iterative Wiener filtering improves speech quality and intelligibility. Iterative Wiener is preferred over the conventional Wiener as it improves speech quality and intelligibility by iteratively improving the Wiener filter estimate.

Table 3.1 Performance of Traditional Algorithms at various Noise Types and Levels

Noise Type	Noise Level (dB)	SNR		segSNR		PESQ		STOI		SI-SDR	
		Iterative Wiener	Hybrid Algorithm	Iterative Wiener	Hybrid Algorithm	Iterative Wiener	Hybrid Algorithm	Iterative Wiener	Hybrid Algorithm	Iterative Wiener	Hybrid Algorithm
Washing Machine Noise	-10	24.51	29.69	2.91	6.57	1.68	2.28	0.49	0.67	2.53	4.1
	-5	28.53	31.52	4.56	7.42	1.75	2.35	0.56	0.68	4.43	5.84
	0	31.24	33.27	5.12	8.35	1.84	2.41	0.59	0.69	5.94	6.31
	5	34.65	35.22	6.44	9.38	1.98	2.58	0.62	0.71	8.21	8.58
	10	37.29	38.02	8.29	10.04	2.19	2.61	0.66	0.72	10.16	11.21
	15	39.87	41.67	10.96	11.29	2.42	2.84	0.72	0.73	13.74	14.28
Rainbow Noise	-10	25.76	29.63	3.14	6.36	1.56	2.37	0.41	0.58	2.46	4.24
	-5	29.59	31.52	4.76	6.86	1.71	2.4	0.49	0.6	4.31	5.82
	0	33.14	33.42	5.92	7.26	1.74	2.49	0.59	0.63	5.92	6.62
	5	35.87	36.06	7.65	8.53	1.95	2.6	0.61	0.68	8.94	9.12
	10	37.21	38.03	8.97	9.8	2.13	2.78	0.63	0.69	10.19	10.72
	15	38.68	40.83	11.95	12.65	2.35	3.12	0.67	0.72	12.98	13.87
Babble Noise	-10	28.61	29.6	2.96	6.71	1.54	2.39	0.4	0.64	2.38	4.23
	-5	31.6	32.47	5.14	7.31	1.62	2.41	0.45	0.65	4.25	5.96
	0	34.72	35.34	6.86	8.01	1.75	2.6	0.49	0.67	5.86	6.84
	5	35.86	36.2	8.37	9.39	1.88	2.66	0.52	0.69	8.76	9.21
	10	37.45	38.96	10.29	10.46	2.17	2.74	0.59	0.71	10.86	10.92
	15	38.32	40.74	12.63	13.25	2.24	2.85	0.64	0.72	12.59	13.25
Airport Noise	-10	27.94	29.57	3.68	6.38	1.63	2.14	0.47	0.66	2.51	4.76
	-5	29.43	31.53	6.93	7.27	1.74	2.31	0.54	0.69	4.26	5.57
	0	31.92	33.36	7.54	8.51	1.86	2.47	0.59	0.7	5.75	6.49
	5	33.14	35.17	9.1	9.34	1.97	2.58	0.63	0.71	8.49	8.76
	10	35.42	37.05	10.62	11.25	2.25	2.74	0.66	0.72	10.63	11.28
	15	39.64	41.19	12.76	13.08	2.42	3.03	0.7	0.75	13.82	14.47

Noise Type	Noise Level (dB)	SNR		segSNR		PESQ		STOI		SI-SDR	
		Iterative Wiener	Hybrid Algorithm	Iterative Wiener	Hybrid Algorithm	Iterative Wiener	Hybrid Algorithm	Iterative Wiener	Hybrid Algorithm	Iterative Wiener	Hybrid Algorithm
Jet Plane Noise	-10	24.93	30.09	2.94	6.56	1.72	2.08	0.43	0.62	2.37	4.52
	-5	27.49	32.6	4.88	7.73	1.89	2.2	0.58	0.66	4.43	6.97
	0	32.45	34.36	7.34	8.7	1.96	2.39	0.62	0.68	6.21	7.78
	5	34.16	36.68	9.73	9.97	2.06	2.58	0.65	0.72	10.21	11.15
	10	37.38	38.22	11.65	11.23	2.24	2.75	0.69	0.74	11.96	12.76
	15	38.95	40.94	13.98	14.12	2.46	2.95	0.71	0.76	13.96	14.46
Street Noise	-10	24.98	29.58	3.39	6.13	1.51	2.26	0.37	0.64	2.63	4.61
	-5	26.87	30.86	4.92	7.31	1.63	2.37	0.42	0.67	5.37	7.49
	0	30.96	32.39	6.58	9.07	1.79	2.55	0.47	0.7	6.84	8.33
	5	33.42	34.11	8.93	10.23	1.97	2.76	0.56	0.71	10.16	11.78
	10	36.63	36.91	9.57	11.36	2.18	2.95	0.59	0.73	11.98	12.16
	15	37.97	38.74	12.73	12.92	2.29	3.02	0.63	0.74	13.36	14.98
Train Whistle Noise	-10	26.58	29.71	3.76	6.33	1.68	2.46	0.39	0.63	2.42	5.06
	-5	28.47	31.63	5.34	7.38	1.82	2.48	0.43	0.69	5.16	7.24
	0	32.56	33.49	6.89	8.23	1.96	2.55	0.47	0.7	6.63	9.12
	5	35.1	36.31	9.28	9.85	2.03	2.61	0.51	0.68	9.95	11.19
	10	38.23	39.13	10.13	11.39	2.34	2.86	0.57	0.71	11.77	13.88
	15	39.57	41.82	13.26	13.87	2.48	3.11	0.7	0.75	13.18	14.68
Restaurant Noise	-10	27.24	29.63	3.15	6.9	1.54	2.37	0.42	0.66	2.46	4.19
	-5	28.67	31.49	5.28	7.73	1.65	2.43	0.49	0.68	5.38	6.97
	0	31.26	33.35	7.32	8.02	1.84	2.67	0.52	0.69	6.63	9.87
	5	33.49	34.62	8.51	9.27	1.95	2.9	0.56	0.7	9.85	11.53
	10	34.96	36.89	10.43	11.32	2.2	3	0.58	0.71	11.74	13.63
	15	37.64	39.76	11.76	12.39	2.38	3.05	0.61	0.73	13.19	14.47

As Wavelet Transform provides a multi-resolution analysis and LMS algorithm, being an adaptive filter that continuously adjusts its parameters based on the input data, the hybrid approach achieves better results than the iterative Wiener filter with reduced computational complexity. The performance metrics SNR, segSNR, PESQ, STOI and SI-SDR obtained from the Iterative Wiener Filter and the Hybrid Algorithm are presented in Table 3.1.

3.6 RESULTS AND DISCUSSION

The clean speech signals taken for the study are 400 sentences from the University of Edinburgh, Centre for Speech Technology Research (CSTR), spoken by 84 male and female speakers at a sampling rate of 48kHz. The different noise signals are taken from https://www.ee.columbia.edu/~dpwe/sounds/noise/ to add the background noise to the clean speech to simulate real-world conditions. Each noise signal is 10 seconds long and sampled at 8kHz. Results were obtained through simulation using MATLAB Software version 2020b.

Based on the results obtained from the metrics such as SNR, segSNR, PESQ, STOI and SI-SDR given in Table 3.1, the performance of the Iterative Wiener filter and the hybrid algorithm is observed. Comparing the Iterative Wiener Filter (IWF) with the Traditional Hybrid Algorithm thoroughly examines their capabilities in dealing with various types of noise. The Hybrid Algorithm demonstrates strong performance across multiple noise profiles, notably excelling in scenarios involving washing machine noise, airport noise, train whistle noise, rainbow noise, babble noise, and jet plane noise. However, its effectiveness is moderate for street noise and restaurant noise. On the other hand, the Iterative Wiener Filter showcases significant improvements in performance metrics, particularly with washing machine noise, airport noise, jet plane noise, and train whistle noise. While it exhibits moderate enhancements in handling other noise types, the IWF's adaptability and iterative refinement highlight its potential for precise noise reduction strategies. Despite this, the Hybrid Algorithm's evident superiority across a broader spectrum of noise environments underscores its efficacy. Therefore, compared to the Iterative Wiener Filter, the hybrid algorithm yields good results, which is evident from the performance metrics.

3.7 SUMMARY

In order to improve the quality of speech based on the performance metrics and to overcome the demerits of the conventional algorithms, the implementation of deep learning algorithms is attempted in the next chapter.