3-2023

Unsupervised Learning Algorithm for Noise Suppression and Speech Enhancement Applications

Abdullah Zaini Alsheibi
University of Denver

Unsupervised Learning Algorithm for Noise Suppression and Speech Enhancement Applications

Abstract

Smart and intelligent devices are being integrated more and more into day-to-day life to perform a multitude of tasks. These tasks include, but are not limited to, job automation, smart utility management, etc., with the aim to improve quality of life and to make normal day-to-day chores as effortless as possible. These smart devices may or may not be connected to the internet to accomplish tasks. Additionally, human-machine interaction with such devices may be touch-screen based or based on voice commands. To understand and act upon received voice commands, these devices require to enhance and distinguish the (clean) speech signal from the recorded noisy signal (that is contaminated by interference and background noise). The enhanced speech signal is then analyzed locally or in cloud to extract the command. This speech enhancement task may effectively be achieved if the number of recording microphones is large. But incorporating many microphones is only possible in large and expensive devices. With multiple microphones present, the computational complexity of speech enhancement algorithms is high, along with its power consumption requirements. However, if the device under consideration is small with limited power and computational capabilities, having multiple microphones is not possible. For example, hearing aids and cochlear implant devices. Thus, most of these devices have been developed with a single microphone. As a result of this handicap, developing a speech enhancement algorithm for assisted learning devices with a single microphone, while keeping computational complexity and power consumption of the said algorithm low, is a challenging problem. There has been considerable research to solve this problem with good speech enhancement performance. However, most real-time speech enhancement algorithms lose their effectiveness if the level of noise present in the recorded speech is high. This dissertation deals with this problem, i.e., the objective is to develop a method that enhances performance by reducing the input signal noise level. To this end, it is proposed to include a pre-processing step before applying speech enhancement algorithms. This pre-processing performs noise suppression in the transformed domain by generating an approximation of the noisy signals' short-time Fourier transform. The approximated signal with improved input signal to noise ratio is then used by other speech enhancement algorithms to recover the underlying clean signal. This approximation is performed by using the proposed Block-Principal Component Analysis (Block-PCA) algorithm. To illustrate efficacy of the methodology, a detailed performance analysis under multiple noise types and noise levels is followed, which demonstrates that the inclusion of the pre-processing step improves considerably the performance of speech enhancement algorithms when compared to other approaches with no pre-processing steps.

Document Type

Dissertation

Degree Name

Ph.D.

Department

Electrical Engineering

First Advisor

Kimon P. Valavanis

Second Advisor

Wenzhong (David) Gao

UNSUPERVISED LEARNING ALGORITHM FOR NOISE SUPPRESSION AND SPEECH ENHAMNCEMENT APPLICATIONS

A Dissertation

Presented to

the Faculty of the Daniel Felix Ritchie School of Engineering and Computer Science

University of Denver

In Partial Fulfillment

of the Requirements for the Degree

Doctor of Philosophy

---

by

Abdullah Zaini Alsheibi

March 2023

Advisor: Dr. Kimon P. Valavanis

Author: Abdullah Zaini Alsheibi
Title: UNSUPERVISED LEARNING ALGORITHM FOR NOISE SUPPRESSION AND SPEECH ENHAMNCEMENT APPLICATIONS
Advisor: Dr. Kimon P. Valavanis
Degree Date: March 2023

Abstract

Smart and intelligent devices are being integrated more and more into day-to-day life to perform a multitude of tasks. These tasks include, but are not limited to, job automation, smart utility management, etc., with the aim to improve quality of life and to make normal day-to-day chores as effortless as possible. These smart devices may or may not be connected to the internet to accomplish tasks. Additionally, human-machine interaction with such devices may be touch-screen based or based on voice commands. To understand and act upon received voice commands, these devices require to enhance and distinguish the (clean) speech signal from the recorded noisy signal (that is contaminated by interference and background noise). The enhanced speech signal is then analyzed locally or in cloud to extract the command. This speech enhancement task may effectively be achieved if the number of recording microphones is large. But incorporating many microphones is only possible in large and expensive devices. With multiple microphones present, the computational complexity of speech enhancement algorithms is high, along with its power consumption requirements. However, if the device under consideration is small with limited power and computational capabilities, having multiple microphones is not possible. For example, hearing aids and cochlear implant devices. Thus, most of these devices have been developed with a single microphone. As a result of this handicap, developing a speech enhancement algorithm for assisted learning devices with a single microphone, while keeping computational complexity and power consumption of the said

algorithm low, is a challenging problem. There has been considerable research to solve this problem with good speech enhancement performance. However, most real-time speech enhancement algorithms lose their effectiveness if the level of noise present in the recorded speech is high. This dissertation deals with this problem, i.e., the objective is to develop a method that enhances performance by reducing the input signal noise level. To this end, it is proposed to include a pre-processing step before applying speech enhancement algorithms. This pre-processing performs noise suppression in the transformed domain by generating an approximation of the noisy signals' short-time Fourier transform. The approximated signal with improved input signal to noise ratio is then used by other speech enhancement algorithms to recover the underlying clean signal. This approximation is performed by using the proposed Block-Principal Component Analysis (Block-PCA) algorithm. To illustrate efficacy of the methodology, a detailed performance analysis under multiple noise types and noise levels is followed, which demonstrates that the inclusion of the pre-processing step improves considerably the performance of speech enhancement algorithms when compared to other approaches with no pre-processing steps.

Keywords: speech and audio enhancement, noise reduction, principal component analysis, MMSE filtering, Block-PCA

Acknowledgments

It was a privilege to work with my advisor Dr. Kimon P. Valavanis. It was an amazing and unforgettable experience in my life. He was a very supportive and helpful professor. Several challenges and obstacles were overcome because of his support and guidance. Here it is the time to thank him for all his advising and assistance, which really helped me in my research and improved my knowledge in my research area. Also, I want to thank my PhD committee members, Dr. Mohammed Abdul-Matin, Dr. Wenzhong Gao, and Dr. Frederic Latremoliere for serving as my committee members and for their time, brilliant comments, and suggestions.

Moreover, I would like to thank all my colleagues in the UTM Robotic-Lab, especially Syed Muhammad Fasih ur Rehman, and Hasan Alqaraghuli as they have provided me with all support and information, which assisted me in my research during my PhD. I also want to extend my deep thanks to the Saudi Arabian ministry of education that granted me this chance to pursue my PhD by supporting me financially.

Finally, I dedicate this work to my wonderful parents (Zaini and Abeer), my lovely wife (Razan) and my kids (Abeer, Hamzah, and Yasmeen) as they assisted and supported me morally and emotionally to finish my degree and overcame all challenges.

Table of Contents

List of Tables

List of Figures

Chapter One: Introduction

1.1 Background

In the signal processing field, one of the most researched, important, and crucial tasks is signal denoising, i.e., (for the case of speech signals) the removal or separation of speech from the noisy received signal. This is carried out by performing speech enhancement, noise signal suppression, or a combination of both. With the widespread acceptance of smart home devices, mobile speech processing applications, assisted listening, and wireless networks enabled by voice communication-based human-machine interface technologies, the need to improve intelligibility and overall quality of the speech signal has become indispensable. The environment in which these applications are used are often noisy and non-stationary, i.e., inside a restaurant, train station, airport, or inside of a moving vehicle. The speech signal recorded in such environments are inherently noisy and are not suitable for systems performing speech coding or command recognition, typically employed by the telephony [1]. Thus, a pre-processing step involving speech enhancement technique is crucial to attain acceptable performance of these systems. Additionally, these speech enhancement techniques are also suited for performing noise reduction before amplification in hearing aids and cochlear implant devices required by audibly impaired listeners.

The main objective of a speech enhancement system is to improve the listeners comfort level and reduce fatigue. To do so, just removing noise from the signal, hence improving quality, is not enough, these methods need to improve intelligibility of the enhanced signal as well. To this end, many solutions have been proposed to perform noise removal for speech enhancement, however, the enhanced speech signal recovered by these methods contains speech distortion. As a result, these algorithms are not only required to reduce noise, but to do so while keeping the speech distortion to a minimum. Moreover, various factors affect the algorithm design for speech enhancement, which are, resource database, type of noise contamination, nature of noise, speech and noise signal relationship, and the number of channels available. Depending on the number of channels/microphones present in the device, speech enhancement methods are categorized into single channel and multi-channel techniques. Simply, the quality of the enhanced signal has a direct relationship with the number of channels used. For example, the microphone placed close to the noise source could better estimate noise. However, in the case of small-scale devices, like hearing aids, size and cost limitations can hinder inclusion of multiple microphones. Additionally, computational complexity and power consumption issues require consideration as well. Similarly pre-recorded single-channel audio streams also cannot benefit from multi-channel techniques. As a result, single channel speech enhancement techniques are employed in the pre-processing step in low cost and small-scale devices. Moreover, single-channel techniques can be used in multi-channel systems after performing spatial filtering or beamforming on the microphones array. In fact, for an additive white Gaussian noise (AWGN) model, an optimal method for multi-channel noise

reduction is a combination of a minimum-variance distortion-less multi-channel beamformer followed by a single-channel noise suppression algorithm [2].

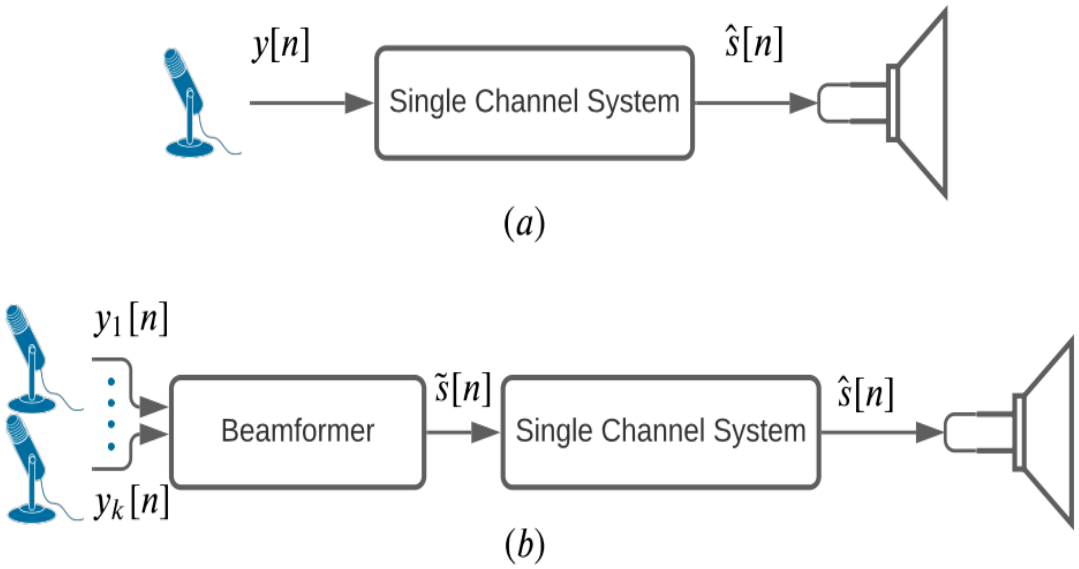


Figure 1.1: (a) a single channel denoising system, (b) a multi-channel system using a single channel denoising system

An example of such a system is given in Figure. 1.1, highlighting the central role of single-channel techniques in both scenarios. In this dissertation, we only consider single-channel speech signals as they are the most challenging and central for speech enhancement.

In what follows, the problem statement, research significance, objectives and the dissertation outline are presented. For clarification purposes, a brief overview of single channel denoising methods, including unsupervised and supervised methods is given. This provides the rationale and motivation for the conducted research that focuses on unsupervised learning methods.

1.2 Problem Statement

Although denoising techniques developed using supervised machine learning approaches have led to good denoising performance, they are not suited for every possible scenario. In such cases, where large training databases for different noise types like speech, and audio signals are not available, supervised learning approaches cannot be effectively applied. In applications where low cost, low complexity, and real-time processing are required to boost sound quality and comfort, unsupervised learning techniques are a better fit. These applications include hearing aids for audibly impaired people and voice communication- and recognition- based products. As a result, in this dissertation, the choice is to focus on unsupervised learning denoising techniques.

Without any prior signal or noise training, unsupervised learning methods that are developed for speech denoising, must satisfy the following requirements (that are considered in this dissertation) to be applicable:

- Perform well for both audio and speech signals in the presence of noise.
- Have a good balance between quality and intelligibility for audio and speech signals.
- Perform well in both stationary and non-stationary noise types and scenarios.
- The response time and computational complexity must be low for real-time applications.

Note that the main key issue with unsupervised learning techniques that have been proposed so far is their susceptibility to the level and amount of noise contamination present in the input speech or audio signal, i.e., under moderate to high signal to noise ratio (SNR), their denoising performance is objectively better. However, under low to poor SNR

conditions, such techniques may fail completely. However, this dissertation aims to remedy this challenge. To do so, a pre-processing step is proposed, which is implemented in the frequency domain that reduces the overall signal noise level; as a result, this leads to an improved input signal SNR. This step creates an approximation of the input signal spectrum in the transform domain with reduced noise power. This pre-processed approximate signal can then be passed to any speech enhancement algorithm, and results in superior performance as compared to cases where the pre-processing step is not included. This pre-processing step includes a variation of the well-known principal component analysis (PCA) technique, named Block-PCA, which operates on blocks of short-time Fourier transform (STFT) of the noisy signal to generate an STFT approximation where the present noise level in the signal has been suppressed.

1.3 Objectives

The main objectives of this dissertation are:

i. Develop a computationally lightweight and power efficient speech enhancement framework for single microphone-based systems, which can be used in low cost and low complexity devices to improve speech enhancement performance.

ii. Introduce a signal pre-processing step in the overall speech enhancement framework to improve the input SNR. This is achieved by using a variation of the dimensionality reduction technique, called block - principal component analysis. This pre-processing step is ‘added’ before the main speech enhancement algorithm leading to improved and enhanced performance (compared to similar approaches).

iii. Evaluate performance using publicly available datasets containing both male and female voice recordings along with noise contamination from multiple real life noisy environments.

1.4 Research Significance

Dissertation research has a wide spectrum of applications. A sample, not exclusive list, of potential applications is:

a. Lightweight speech assisted hearing aids and cochlear implants for voice impaired individuals.
b. Power and computationally efficient speech recognition systems.
c. Mobile telephony and teleconference systems.

1.5 Dissertation Organization

This dissertation is organized as follows. Chapter 2 offers a literature review. Chapter 3 presents the proposed methodology, followed by the principal component analysis (PCA) technique. It also presents a detailed simulation example to further illustrate the motivation behind using PCA to perform noise suppression. The proposed pre-processing step is also detailed along with the complete implementation framework. Experimental validation and performance evaluation is provided in Chapter 4. Conclusions and discussion on future work are presented in Chapter 5. It is also stated that the results of this research have been published in [80].

Chapter Two: Literature Review

Considerable research has been conducted in the topic of single-channel speech enhancement. These methods can be broadly categorized into two classes: unsupervised and supervised methods. In supervised methods, the system is first trained using a large enough database to learn properties and features of speech as well as noise signal, then using this acquired knowledge, the system proceeds to clean the input noisy signal. Unsupervised methods, on the other hand, do not require such training, they use statistical tools to estimate the noise properties and try to remove the noise contamination from the noisy signal, leading to a signal with improved speech quality. Here we provide a brief overview of the classes along with some basic discussion about their sub-classes as well.

2.1 Unsupervised Learning Methods

This class of speech enhancement methods is rather rich with their primary aim being improvement in speech intelligibility and quality. Authors in [1, 3] have provided a detailed review of the topic, where most of the techniques operate in signal Frequency domain, using Discrete Fourier Transform (DFT) [4]. These methods can be further distinguished as parametric and non-parametric techniques. In short, parametric techniques are those where some sort of prior information about signal or noise distributions is, up to some certainty, available, which is then utilized via standard Bayesian and Likelihood theory. In contrast, non-parametric methods assume no such knowledge about the signal distribution.

2.1.1 Parametric Methods

These methods assume the distributions of clean and noise signal to be known a-priori. These methods perform clean signal estimation by formulating the signal denoising framework using maximum likelihood (ML) [5], maximum a-posteriori (MAP) [6, 7], or minimum mean square error (MMSE) [8, 9]. MAP and MMSE estimators require the knowledge about probability distribution function (PDF) of speech, in [8], speech pdf is assumed to be Gaussian, super-Gaussian in [10, 11], Laplacian in [12], and generalized gamma distribution in [13]. The cost function used in MMSE estimators are mean-square magnitude error (ℓ_2–norm), log-magnitude spectra, or other distortion measures like Itakura-Saito or Cosh measures [14]. In addition, in most of these methods, noise distribution is assumed to be Gaussian or Laplacian [15]. Moreover, some methods also use the voice activity detectors (VAD) to better estimate noise and speech distribution, leading to further improvement in speech quality [16, 17, 18].

2.1.2 Non-Parametric Methods

Under this framework, the simplest and most computationally efficient are power spectral subtraction-based algorithms [19, 20, 21, 22, 23]. These algorithms do not require much prior information about speech and noise signals and use a basic additive noise model. Another set of techniques is the optimal Wiener algorithm, which performs the speech enhancement by assuming the relationship between clean signal coefficients and noisy coefficients to be linear [24, 25, 26, 27, 28]. Subspace decomposition is another subclass, whose goal is to decompose the noisy signal into clean signal subspace and noisy-only subspace [29, 30, 31, 32, 33]. Furthermore, some binary masking algorithms have also

been proposed for speech intelligibility improvement [34, 35, 36]. These techniques work by keeping only a few frequency bins from the noise spectra while forcing the rest to zero, hereby, enabling noise suppression.

2.2 Supervised Learning Methods

In supervised methods, training is carried out using clean speech signals and noisy signals separately to learn the model parameters. Once the model parameters are learned, noisy signal is decomposed into clean and noisy only signal. These methods can be divided into four sub-classes, i.e., Hidden Markov Model (HMM) based methods, codebook-based Wiener filtering methods, sparse representation-based methods, and Deep Neural Networks (DNN) based methods.

2.2.1 Codebook-based Wiener Filter Methods

The noise model used here is linear and additive in nature. Based on the Wiener filter, methods in this class use codebooks of auto-regressive (AR) parameters to perform linear prediction synthesis of the noise and speech signals. The Wiener filter is essentially the ratio of clean signal spectra and noise power spectra. These methods assume that the noise power spectrum is the sum of clean signal spectra and noise spectra, which can be estimated by the AR parameters. Based on this assumption, these methods use a training database containing clean signal and noise examples to learn codebooks for speech and noise spectra. The training for both spectra can be performed offline [37, 38, 39]. Alternatively, in [40], learning signal spectra is done offline and noise spectra is learned online. The estimation of observed signal's AR coefficients is performed by Bayesian MMSE or ML based criterion using the learned codebooks.

2.2.2 HMM-based Methods

In HMM-based methods, the clean signal and noise AR parameters are modeled via HMM instead of linear prediction synthesis approach used in codebook-based methods. In [41, 42, 43], authors assume Gaussian AR parameters for both noise and speech signals, whereas authors in [44, 45] assumed the signal coefficients to have complex Gaussian or super Gaussian distribution in the transformed domain and worked with them in the transformed domain directly. Expectation Maximization (EM) algorithm is used to train the model parameters using speech and noise database. Finally, the clean speech signal is estimated using Bayesian MMSE or MAP estimators from the noisy signal based on the trained model parameters. The transformed domain can be Fourier (computed efficiently using discrete Fourier transform (DFT)) or reduced resolution frequency domain.

2.2.3 DNN-based Methods

Deep neural networks have been intensively used in several applications and perform very well if the training data is sufficiently large. First application of DNN methods for speech enhancement was presented in year 2013 [46]. Like other supervised approaches, speech enhancement using DNN is carried out in two stages. In the first stage, the training stage, clean signal, and noise samples are used to learn the parameters of interest (log-amplitude and phase) of clean and noisy signals in the DFT domain [46, 47, 48]. Training is performed using a regression DNN model. In the second stage, enhancement stage, noisy signal is given to the trained DNN, and clean signal amplitude is estimated. This stage is followed by a post-processor stage which brings further improvement in the speech quality [47, 48].

2.2.4 Sparse Representation based Methods

These methods are like Codebook-based Wiener Filter methods in the sense that both these methods try to learn a codebook containing the parameters of interest. However, the way these codebooks are learned is entirely different. Sparse representation-based methods learn a codebook named Dictionary using the assumption that the signals of interest belong a an underlying overcomplete union of subspaces (called Dictionary), in which, every signal can be sparsely represented by only a few bases (columns of the dictionary) [49, 50, 51, 52]. Once the dictionary is trained, it is then used in the enhancement stage, along with the noisy signal to estimate the noise-free speech signal. Typically, a Wiener filter-type or MMSE estimators are used for the said process.

2.3 Speech Enhancement Frameworks

Before discussing the proposed methodology for speech denoising, in this chapter, we present a detailed overview of a typical single channel system for speech enhancement and the performance metrics used to assess such systems. This chapter is divided into three sections. In section 2.3.1, we discuss the general architecture of a single channel speech enhancement system. In section 2.3.2, various performance metrics for such a system are presented. Section 2.4 summarizes the chapter.

2.3.1 Single Channel Speech Enhancement System

The removal or reduction in noise amplitude from an input noisy signal is the most crucial task of any speech enhancement algorithm. Typically, the input noisy signal is segmented using an appropriate window and is transformed into a parallel representation domain. In this domain, speech enhancement algorithms try to estimate the clean signal

coefficients from the noisy transformed observations. Finally, the enhanced signal is recovered by transforming back into the temporal signal domain. A typical single channel speech enhancement system [1], shown in Figure 2.1, consists of four blocks: signal decomposition, noise estimation, noise reduction, and signal reconstruction block. The entire process is explained as follows:

1. The decomposition block performs two tasks:
    - Decomposes the 1D noisy signal samples ***y[n]*** into multiple overlapping segments using a window signal ***w[n]***.
    - Takes a short time harmonic transform (STHT) on the time axis to generate time-frequency noisy signal $\boldsymbol{Y}[k,m]$.

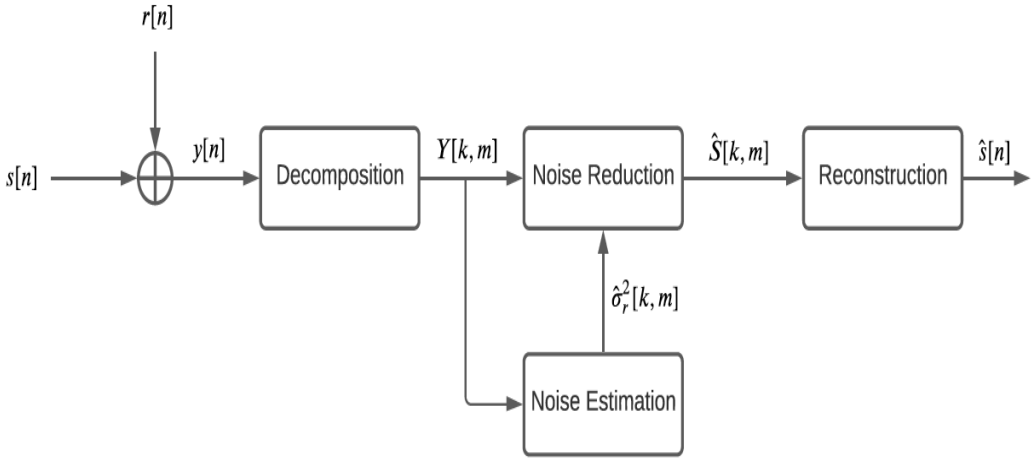


Figure 2.1: A typical single channel audio enhancement system showing clean discrete-time signal $\boldsymbol{s}[n]$, noise contamination $\boldsymbol{r}[n]$, noisy signal $\boldsymbol{y}[n]$, segmented and transformed noisy signal $\boldsymbol{Y}[k,m]$, noise power spectrum $\hat{\sigma}_r^2[k,m]$, enhanced signal coefficients $\widehat{\boldsymbol{S}}[k,m]$, and reconstructed clean time-domain signal $\hat{\boldsymbol{s}}[n]$

2. The noise estimation block estimates the noise power spectrum $\hat{\sigma}_r^2[k,m]$ from the transformed noisy signal $\boldsymbol{Y}[k,m]$ and forwards it to the noise reduction block.
3. The noise reduction block uses the noise power spectrum $\hat{\sigma}_r^2[k,m]$ and estimates the enhanced signal coefficients $\widehat{\boldsymbol{S}}[k,m]$.
4. The reconstruction block uses the time-frequency enhanced coefficients $\widehat{\boldsymbol{S}}[k,m]$, computes its inverse STHT, and synthesizes the enhanced time-domain speech signal $\hat{\boldsymbol{s}}[n]$ using typical overlap and add method.

In this dissertation, the Hamming window is used, and an overlap of 50% in all algorithmic implementations. Additional implementation details of each block are given next.

The Decomposition Block

Most of the speech enhancement algorithms have been implemented in transformed domain, rather than time domain, that is due to the fact that the separation of signal of interest and noise is easier to accomplish in the transformed domain. Our proposed pre-processing method also works in the transformed domain. As discussed earlier, the decomposition block performs this transformation. Let $\boldsymbol{s}[n]$ be the clean signal of interest and $\boldsymbol{r}[n]$ be the uncorrelated noise signal, we consider the noise contamination process to be additive, thus the received noisy signal is modeled as $\boldsymbol{y}[n] = \boldsymbol{s}[n] + \boldsymbol{r}[n]$. These signals are currently assumed to be sampled in time domain, with n = 1, 2, . . ., *N* being the sample time index. The real-time enhancement algorithms operate on chunks of the signal, called frames, instead of the whole sampled signal. This contrasts with offline algorithms where the entire signal stream is available. These frames are generated by decomposing the signal stream using a suitable window function $\boldsymbol{w}_L[n]$ of length *L*.

$$\boldsymbol{y}_w[n] = \boldsymbol{y}[n+n_0]w_L[n] = \begin{cases} 0 & n < 0 \\ \boldsymbol{w}_L[n]\,\boldsymbol{y}[n+n_0] & 0 \le n \le L-1 \\ 0 & n \ge L \end{cases}$$

(2.1)

In (2.1), the output $\boldsymbol{y}_w[n]$ contains L values of $\boldsymbol{y}[n]$ starting from $n = n_0$, i.e., the changes in n_0 produce different signal shifts, and we see a different length L frame of the signal. Three example window functions are given in Figure. 2.2. Based on the frequency responses of these window functions, we chose to work with Hamming window as spectral leakage caused by this window is vastly lower than the others because its samples do not vanish to zero near the end. In addition to generating frames, these windows also act as narrowband low pass filters, contributing to the reduction of spectral leakage [53]. The length L of the window allows us to control the trade-off between statistical variance and spectral resolution. Proper care is required to set this parameter, as a small length leads to less spectral accuracy, whereas large length may lead to non-stationarity of the signal of interest within frame. The Hamming window function is given below:

$$\boldsymbol{w}[n] = \begin{cases} 0.54 - 0.46\,cos\left(\dfrac{2\pi n}{L-1}\right) & 0 \le n \le L-1 \\ 0 & Otherwise \end{cases}$$

(2.2)

with L representing the window length.

Once the segmentation is complete, the resulting segmented noisy signal can be written as

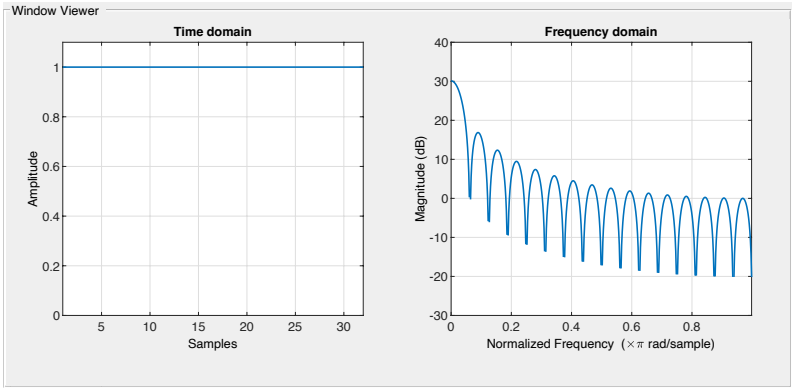
$$\boldsymbol{y}_w[n] = \boldsymbol{y}[n,m] = \boldsymbol{s}[n,m] + \boldsymbol{r}[n,m]$$

(2.3)

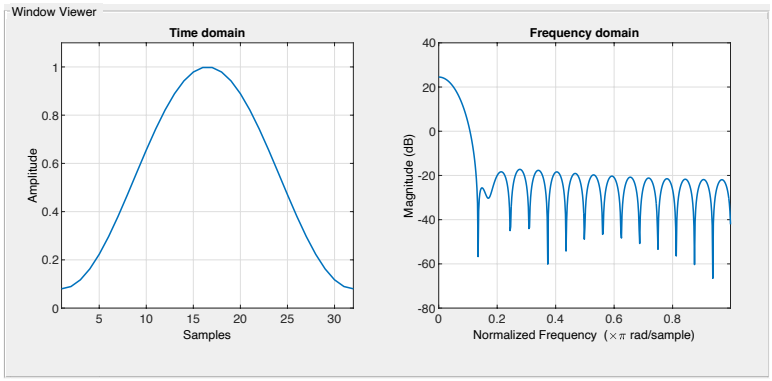
where n represents the time points, and m representing the segment/frame number. The next operation in the pipeline is the application of short time transform to these signal frames. The transforms useful here are wavelet transform, discrete cosine transform, and Fourier transform. Once the transformation is complete, the resulting signal is given by

$$\boldsymbol{Y}[k,m] = \boldsymbol{S}[k,m] + \boldsymbol{R}[k,m] \tag{2.4}$$

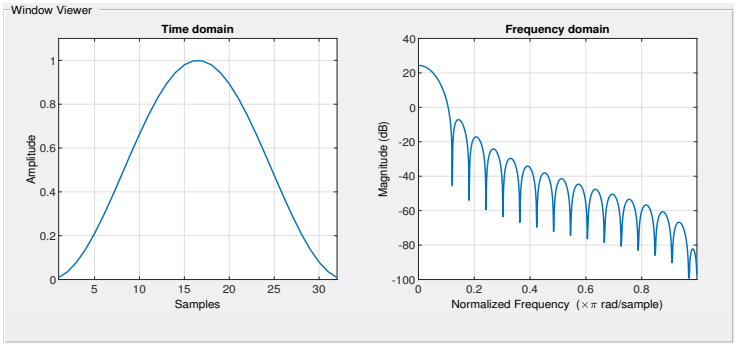
where $k = 0,1, \ldots \ldots \ldots, L-1$ is the frequency sample index. The computation is given



(a) Rectangular window



(b) Hamming window



(c) Hamming window

Figure 2.2: Window samples and frequency response for a) Rectangular window, b) Hamming window, and c) Hanning window of size 32.

below

$$\boldsymbol{Y}[k,m] = \sum_{n=0}^{L-1} \zeta_k[n]\boldsymbol{y}[n,m] \tag{2.5}$$

Where $\zeta_k[n]$ is the transform dependent basis function. For the case of short time Fourier transform (STFT), this function is written as

$$\zeta_k[n] = exp(-j(\frac{2\pi}{L})kn) \tag{2.6}$$

with $2\pi/L$ frequency bin width. Thus, the larger window length, more detailed spectrum can be obtained. The resulting transform is complex-valued spectrum with magnitude and phase spectra, i.e.,

$$\boldsymbol{Y}[k,m] = \boldsymbol{P_Y}[k,m]\, exp(-j\theta_{\boldsymbol{Y}}[k,m] \tag{2.7}$$

where $\boldsymbol{P_Y}[k,m]$ is the amplitude spectra and $exp(-j\theta_{\boldsymbol{Y}}[k,m])$ is the phase spectra of the noisy signal of each frame. As a result, the frequency domain signal model (2.4) is given by

$$\boldsymbol{P_Y}[k,m]\, exp(-j\theta_{\boldsymbol{Y}}[k,m]) = \boldsymbol{P_S}[k,m]\, exp(-j\theta_{\boldsymbol{S}}[k,m]) + \boldsymbol{P_R}[k,m]\, exp(-j\theta_{\boldsymbol{R}}[k,m]) \tag{2.8}$$

where $\boldsymbol{P_s}[k,m]$ and $exp(-j\theta_{\boldsymbol{S}}[k,m])$ represent the amplitude and phase spectra of clean speech signal, and $\boldsymbol{P_R}[k,m]$ and $exp(-j\theta_{\boldsymbol{R}}[k,m])$ represent the amplitude and phase spectra of noise signal respectively.

Noise Estimation Block

The major step, and the most crucial one, in this block is the estimation of a priori SNR. If this estimation is not accurate enough, the enhancement process introduces audible speech distortion and musical noise into the resulting enhanced speech signal. The decision directed (DD) approach, presented in [8], is a state-of-the-art a priori SNR estimation method. This method also avoids the introduction of musical noise into the enhanced signal. This method uses a weighted averaging scheme by combining the magnitude spectrum estimate of previous frame and current frames' maximum likelihood estimate (MLE) of the a priori SNR. This scheme can be defined as

$$\hat{\gamma}[k,m] = \beta \frac{\left|\widehat{\boldsymbol{S}}[k,m-1]\right|^2}{\hat{\zeta}_r[k,m-1]} + (1-\beta)\left([\hat{\lambda}[k,m]-1)\right)_+ \tag{2.9}$$

Where $\left|\widehat{\boldsymbol{S}}[k,m-1]\right|^2$ is the magnitude estimate and $\hat{\zeta}_r[k,m-1]$ is the noise estimate at the previous frame respectively. $(.)_+$ is a function to keep the argument value non-negative, and $0 < \beta < 1$ is the weighting factor controlling the contribution of a priori SNR estimate of the previous frame and the posterior SNR estimate of the current frame in the current frames' a priori SNR estimate.

The assigned β value in (2.9) is the key to an accurate calculation of a priori SNR, and thus its setting is key. In [54], authors have discussed two different behaviors of the estimated a priori SNR when setting β close to 1. In this case, for noise frames with posterior SNR estimate is lower or close to 0 dB, the resulting a priori SNR estimate for the current frame is equal to a scaled version of posterior SNR as the second term in (2.9) approaches zero. By putting (2.17) with $\widehat{\boldsymbol{S}}[k,m] = \boldsymbol{G}[k,m]\boldsymbol{Y}[k,m]$ into (2.9), the a priori SNR estimation becomes

$$\hat{\gamma}_{DD}^{\downarrow}[k,m] \approx \beta \boldsymbol{G}^2[k,m-1]\hat{\lambda}[k,m-1] \tag{2.10}$$

The $\downarrow$ corresponds to the case where no speech signal is present in the frame under consideration. As a result of this, the a priori SNR estimates' variations are minimized, which in turn, leads to reduction in intensity of the produced musical noise. On the other hand, for frames containing speech onsets, the a priori SNR follows closely the previous frames' posterior SNR according to the following relationship

$$\begin{aligned}\hat{\gamma}_{DD}^{\uparrow}[k,m] &= \beta \frac{\boldsymbol{G}^2[k,m-1]|\boldsymbol{Y}[k,m-1]|^2}{\hat{\zeta}_r[k,m-1]} + (1-\beta)([\hat{\lambda}[k,m]-1])_+ \\ &\approx \beta \boldsymbol{G}^2[k,m-1]\hat{\lambda}[k,m-1] + (1-\beta)\,([\hat{\lambda}[k,m]-1])_+\end{aligned} \tag{2.11}$$

With $\beta \approx 1$, the second term (the ML estimate) will have very little affect on the overall estimation process. Furthermore, the variation in the a priori SNR estimate is slow as its solution mainly depends on the previous frames' posterior SNR estimation. The introduction of speech distortion into the signal is a consequence of this behavior. To remedy this problem, a modified decision directed (MDD) approach has been presented in [55]. In this method, instead of matching the posterior SNR of previous frame, the posterior SNR of current frame is matched with current frames' a priori SNR estimate. This results in a single frame delay as compared to the DD approach leading to reduction in speech distortion. The a priori SNR estimate under MDD is given as

$$\hat{\gamma}_{DD}[k,m] = \beta \frac{\boldsymbol{G}^2[k,m-1]|\boldsymbol{Y}[k,m]|^2}{\hat{\zeta}_r[k,m]} + (1-\beta)([\hat{\lambda}[k,m]-1])_+ \tag{2.12}$$

Noise Reduction Block

Typically, the main task of a speech enhancement algorithm is to extract the enhanced speech signal $\widehat{\boldsymbol{S}}$ by utilizing a specific spectral gain function $\boldsymbol{G}(m)$, i.e., gain function of the m^{th} frame. This gain function is applied to spectrum of the noisy signal to get the enhanced speech signal.

Several gain functions have been proposed. Typical gain functions are computed based on the a priori SNR ($\gamma[k,m]$). As an example, consider the Weiner filter (WF) based gain function [56] defined as

$$\boldsymbol{G}_{WF}[k,m] = \frac{\gamma[k,m]}{1+\gamma[k,m]} \tag{2.13}$$

With

$$\gamma[k,m] = \frac{\zeta_s[k,m]}{\zeta_r[k,m]} \tag{2.14}$$

where $\zeta_s[k,m] = E[|\boldsymbol{S}[k,m]|^2]$ and $\zeta_r[k,m] = E[|\boldsymbol{R}[k,m]|^2]$ represent the power spectral densities of clean speech signal and noise signal, respectively.

Another popular speech estimator is the MMSE-LSA [9], which is the result of minimization of mean square error of the logarithm of enhanced and noisy speech spectrum. The resulting gain function is a function of the posterior SNR $\lambda[k,m]$, and a prior SNR $\gamma[k,m]$, and is given by

$$\boldsymbol{G}_{MMSE}[k,m] = \frac{\gamma[k,m]}{1+\gamma[k,m]} exp\left\{\frac{1}{2}\int_{\delta_k}^{\infty} \frac{e^{-t}}{t} dt\right\} \tag{2.15}$$

Where the δ_k is

$$\delta_k = \frac{\gamma[k,m]}{1+\gamma[k,m]} \lambda[k,m] \tag{2.16}$$

and the posterior SNR is computed as

$$\lambda[k,m] = \frac{|\boldsymbol{Y}[k,m]|^2}{\zeta_r[k,m]} \tag{2.17}$$

As discussed earlier, most of the speech enhancement algorithms use a gain function $\boldsymbol{G}[k,m]$ to compute the enhanced speech amplitude spectrum from the noisy signal spectrum as follows:

$$\widehat{\boldsymbol{P}}S[k,m] = \boldsymbol{G}[k,m]\boldsymbol{P}_Y[k,m] \tag{2.18}$$

On the other hand, the enhanced phase spectra $\hat{\theta}_{\boldsymbol{S}}[k,m]$ is set to the noisy signals' phase spectra $\theta_{\boldsymbol{R}}[k,m]$. Thus, the complex-valued transformed domain estimated coefficients $\widehat{\boldsymbol{S}}[k,m]$ is given by

$$\widehat{\boldsymbol{S}}[k,m] = \widehat{\boldsymbol{P}}_{\boldsymbol{s}}[k,m]\ exp(j\hat{\theta}_{\boldsymbol{s}}[k,m]) = \boldsymbol{G}[k,m]\boldsymbol{Y}[k,m] \tag{2.19}$$

To summarize, the noise reduction block performs the estimation of enhanced transformed domain signal coefficients $\widehat{\boldsymbol{S}}[k,m]$ by multiplying the gain function $\boldsymbol{G}[k,m]$ to the noisy transformed domain signal coefficients $\boldsymbol{Y}[k,m]$. The gain function is computed from the estimated a priori SNR and the posterior SNR estimated in the noise estimation blocks.

Reconstruction Block

This block is the final step in the speech enhancement process. Here the enhanced transformed domain signal coefficients $\widehat{\boldsymbol{S}}[k,m]$ are transformed back into the time domain. As the STFT is an invertible transform, the exact signal reconstruction from time - frequency and back to time is possible. There are several algorithms in the literature which can be used to accomplish this reconstruction [57, 58, 59]. However, in this section we

present the implementation of overlap-add method [58] which is the most frequently used method for such reconstruction. For a given frame m, the time enhanced signal $\hat{\boldsymbol{s}}_m[n]$ is given by

$$\hat{\boldsymbol{s}}_m[n] = \sum_{k=0}^{L-1} \alpha_n[k]\widehat{\boldsymbol{S}}[k,m] \tag{2.20}$$

In case of short time Fourier transform, the $\alpha_n[k]$ is given by

$$\alpha_n[k] = exp(j(\frac{2\pi}{L})kn) \tag{2.21}$$

Once we obtain all overlapped frames for $\hat{\boldsymbol{s}}_m[n]$, the final enhanced signal $\hat{\boldsymbol{s}}[n]$ is calculated as

$$\hat{\boldsymbol{s}}[n] = \sum_{k,m} \hat{\boldsymbol{s}}_m[k] \tag{3.22}$$

where $k = 0,1,\ldots,L-1$. To illustrate the overlap and add scheme, consider the illustration shown in Figure. 2.3. Here, to reconstruct the enhanced signal $\hat{\boldsymbol{s}}[n]$ from different frames $\hat{\boldsymbol{s}}_m[n]$, for the case of 50% overlap, we need only 1 previous frame, whereas, in 75% overlapped case, we need previous 3 frames to reconstruct the current signal samples.

In addition to Figure. 2.3, we also give an example of a signal $x(t) = cos(14\pi t) + 2\,cos(20\pi t)$ using 50% overlap as well as 75% overlapping frames in Figure 2.4. The sampling rate was 8000 samples per second and the signal duration was 1 second.

We took STFT of the original signal (shown in blue), using a 1024 sampled Hamming window with a specific overlap ratio, and reconstructed the samples by taking the ISTFT and using the overlap and add method. The resulting reconstructed signals are shown in orange. We can see that as compared to 50% overlap version, the 75% overlapped frames resulted in better signal reconstruction, which is intuitive. Moreover, the error at signal onset and ending is large in both cases, which is predictable as well.

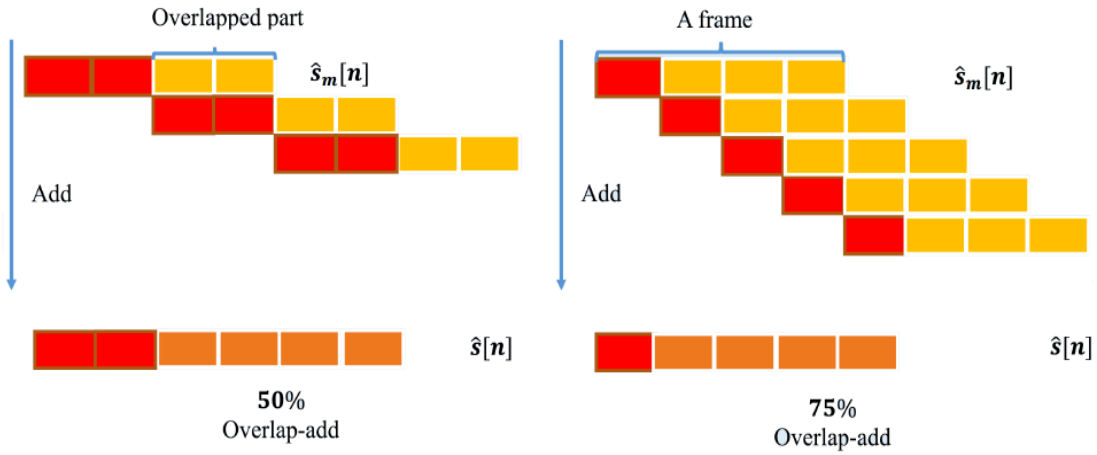


Figure 2.3: Overlap and add formulation for two overlap ratios. Left: 50% frame overlap, Right: 75% frame overlap. The overlapping part of the frame across multiple frames has the same samples.

2.3.2 Performance Metrics for Speech Enhancement Methods

Typical speech enhancement systems consist of four main blocks, as shown in Fig.2.1. Now, to perform quantitative performance analysis, the input noisy signal and enhanced output speech signal are compared using different performance metrics. In this section, we will discuss a few of the most frequently used performance metrics in literature. The methods discussed below have also been used in this dissertation to perform quantitative analysis. In small experimental setups, subjective listening tests can also be used for performance evaluation, however, due to the large number of noisy datasets and

low SNRs used in our tests, we chose to only work with quantitive metrics, which are also easier to reproduce. The metrics used in this dissertation are discussed next.

2.3.2.1 Segmental Signal to Noise Ratio

The segmental signal to noise ratio (SSNR) metric is amongst the most frequently used and simplest criteria. It is computed by taking the geometric mean of the SNR over all segments (frames) of the speech signal [1]. Its formulation is given as

$$SSNR = \frac{10}{M}\sum_{m=0}^{M-1} log_{10}\frac{\sum_{n=Lm}^{Lm+L-1}\boldsymbol{s}^2[n]}{\sum_{n=Lm}^{Lm+L-1}(\hat{\boldsymbol{s}}[n]-\boldsymbol{s}[n])^2} \tag{2.23}$$

Here, M represents total frames, L is the frame length, $\boldsymbol{s}[n]$ is the noise free signal, and $\hat{\boldsymbol{s}}[n]$ is the enhanced signal at the systems output. Additionally, the signals being compared need to be synchronized in time and have the same frame size. SSNR values outside the range of $[-10,35]$ dB is not considered here as SSNR values outside this range do not show much perceptual different in terms of sound quality [60].

2.3.2.2 Weighted-Slope Spectral Distance

Another performance metric under consideration is the weighted-slope spectral distance (WSS) [61]. This metric is found by computing the weighted difference amongst the spectral slopes in every frequency band. The spectral slope is computed in dBs and is the difference between adjacent spectral magnitudes. The mathematical formulation used in this dissertation for WSS computation is

$$WSS = \frac{1}{M}\sum_{m=0}^{M-1}\frac{\sum_{K=1}^{L} W[k,m](\boldsymbol{S}[k,m] - \widehat{\boldsymbol{S}}[k,m])^2}{\sum_{K=1}^{L} W[k,m]} \tag{2.24}$$

where $W[k,m]$ are the weights calculated as described in [61]. $\boldsymbol{S}[k,m]$ and $\widehat{\boldsymbol{S}}[k,m]$ are the spectral slopes for m^{th} frequency band at k^{th} frame of the noise free and enhanced speech signals, respectively.

2.3.2.3 Perceptual Evaluation of Speech Quality

The most widely used speech quality metric for evaluating noise reduction performance of algorithms for telephone handset applications is the perceptual evaluation of speech quality measure (PESQ) [62, 1]. Detailed discussion can be found in [1, Sec 11.1.3.3]. The system to compute PESQ metric consists of five sections; preprocessing, time-alignment, auditory transformation, disturbance processing, followed by the time-frequency averaging blocks. Starting with noise free and enhanced speech signal, both are:

1. Passed through a pre-processing system block to adjust the volume levels of both signals in addition to adapting them to a standard telephone handset. These pre-processed signals are, then,
2. Aligned in time by estimating the time delay value between these signals. Additionally, this system block also provides a time delay confidence level.
3. The auditory transformation block encodes the noise free and enhanced signals into a perceptual representation of the perceived loudness, where the loudness spectra of both signals can be distinguished.

4. The disturbance processing block then computes the difference between both signals.
5. Finally, the time-frequency averaging block computes the PESQ measure from the dissimilarity measured in the previous block.

2.3.2.4 Composite Measures

Composite objective metrics were computed by a linear/nonlinear combination of basic objective measures. These composite measures are significant as we cannot expect the conventional log likelihood ratio (LLR) [63] to have high correlation with speech/noise distortions and the overall quality. Various composite measures have been proposed. In this dissertation, we have used the composite metrics based on multivariate adaptive regression splines (MARS) [62] as they have been extensively used in literature due to their ability to capture good correlation with listening tests. The three metrics used in this dissertation are MARSovrl, MARSbak, and MARSsig denoted here as C_{ovl}, C_{bak}, and C_{sig} respectively. Their computation [1] are given below:

$$C_{sig} = 3.093 - 1.029 * \mu_{LLR} + 0.603 * PESQ_{MOS} - 0.009 * WSS \quad (2.25)$$

$$C_{bak} = 1.634 + 0.478 * PESQ_{MOS} - 0.007 * WSS + 0.063 * SSNR \quad (2.26)$$

$$C_{ovl} = 1.594 + 0.805 * PESQ_{MOS} - 0.512 * \mu_{LLR} + 0.007 * WSS \quad (2.27)$$

Here μ_{LLR} is the mean LLR score, and *MOS* is the mean opinion score of the subjective listening test [62]. C_{ovl} predicts the overall quality of the recovered speech signal, C_{bak}, and C_{sig} were designed to provide good correlation with the two subjective

metrics, i.e., background intrusiveness and signal distortion respectively. Additionally, [1,5] is the acceptable range of values for these composite metrics can take.

2.4 Summary

To summarize, in this chapter we outlined a typical speech enhancement system structure for the case of single channel noisy speech input. After discussing the individual system blocks, we discussed the quantitative metrics used in this dissertation to evaluate the performance of a speech enhancement system. These metrics include segmental signal to noise ratio (SSNR), weighted-slope spectral distance (WSS), perceptual evaluation of speech quality (PESQ), and three composite measures C_{ovl}, C_{bak}, and C_{sig}.From chapter 4 we briefly discuss the motivation behind proposing the pre-processing noise suppression framework and provide some background on the principal component analysis (PCA), technique used to perform said noise suppression.

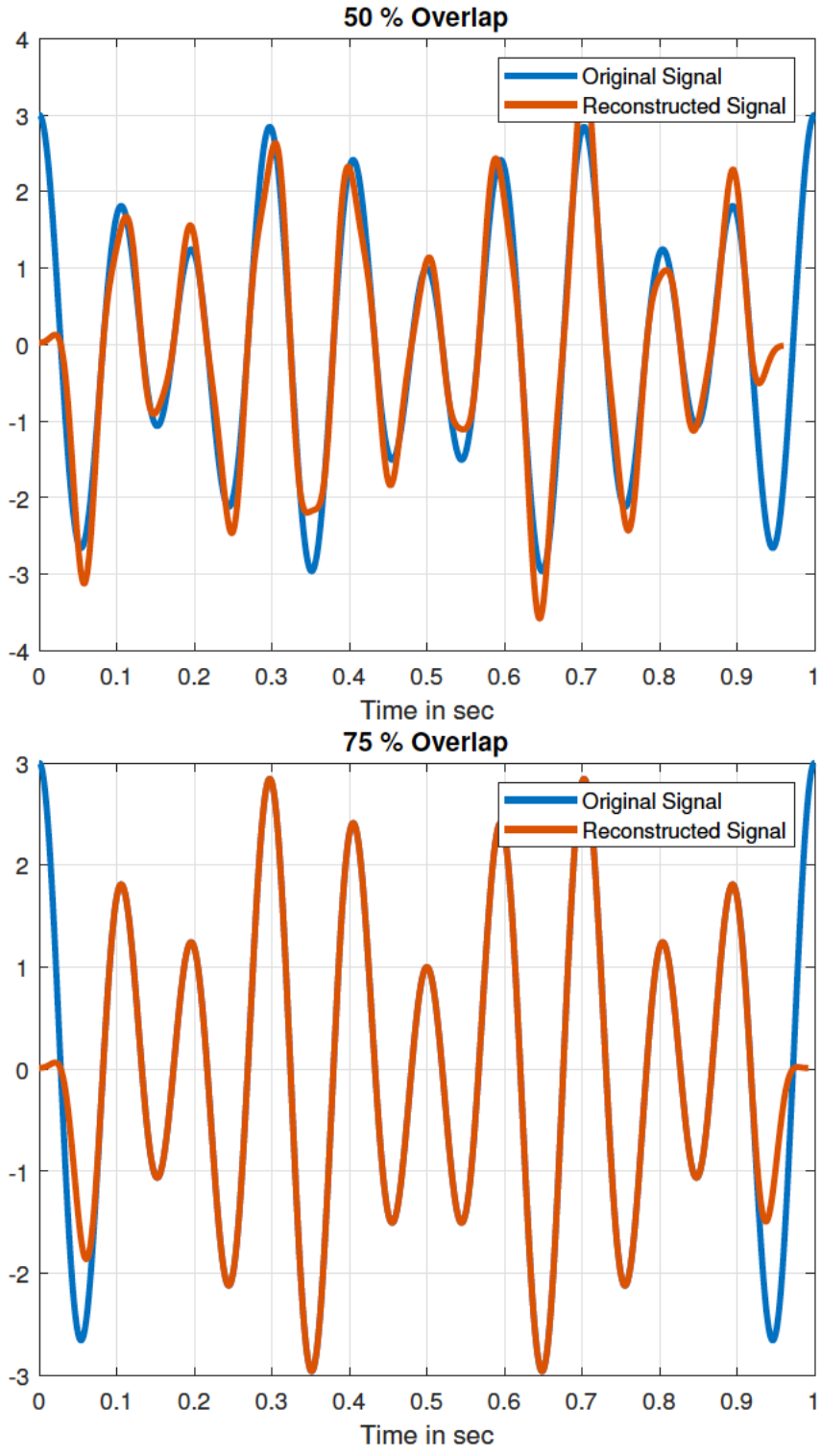


Figure 2.4: Overlap and add signal reconstruction example with 50% (top) and 75% (bot) frame overlap

Chapter three: Research Methodology

3.1 Introduction

The use of multiple recording microphones can help ease the difficulty level of speech enhancement. However, in the case of small-scale devices, like hearing aids, size and cost limitations can hinder inclusion of multiple microphones. Similarly pre-recorded single-channel audio streams also cannot benefit from these techniques. As a result, single-channel speech enhancement techniques, that have low computational cost and low power requirements, are often employed in the pre-processing step in low cost and small-scale devices. These techniques perform either noise estimation or signal estimation (or both) to infer the clean signal from a noisy observation. In [64, 65], voice activity detectors (VAD) are used to approximate frames of signal where the speaker is silent and uses these frames to estimate the noise spectrum. Once the noise estimate is obtained, its spectrum is subtracted from the noisy spectrum, thus reducing the overall noise level. These techniques, however, inherit the shortcomings associated with VAD usage, i.e., they may fail if frame size of the input signal is large, or the signal to noise ratio (SNR) is low.

Similarly, many signal estimation techniques such as spectral subtraction [66, 21], Weiner filtering [67], and minimum mean square error (MMSE) estimation-based methods [8, 9, 68] have been proposed. Out of these, the spectral subtraction methods are the most computationally efficient. They operate by estimating noise spectrum from the noisy speech signal and then recovering the clean signal by subtracting the estimated noise

spectrum from the noisy one. These methods generally perform well, however (the so-called) musical noise might appear in their recovered signal. On the other hand, the MMSE based methods are free from the issue of musical noise. The authors in [8] derive and use a filter using the minimum mean square error - short-time spectral amplitude (MMSE-STSA) estimator based on the cost function minimizing the mean squared error of the short time amplitude of the spectrum. The authors in [9] use a perceptually motivated cost function minimizing the mean squared error between the logarithmic spectral amplitude of clean and enhanced speech signal, whereas [68] propose an adaptive β-order MMSE estimator to design a filter. The designed filter is then used to enhance the signal frequencies leading to a clean speech signal.

One key issue with all these algorithms is their susceptibility to the input SNR levels, i.e., under moderate to high SNR, their enhancement results are objectively better, however, under low to poor SNR conditions, these methods fail altogether. In this manuscript, we aim to remedy this situation by introducing a pre-processing step implemented in the frequency domain which reduces the overall noise level of the signal, resulting in improvement of SNR. This pre-processed signal is then passed to other speech enhancement algorithms as usual, leading to superior results. The pre-processing step involves a variation of the principal component analysis (PCA) technique, termed Block-PCA which operates on blocks of the short-time Fourier transform (STFT) of the noisy signal to generate an STFT approximation by suppressing the noise level present in the signal.

Principal component analysis (PCA) is one of the most utilized feature extraction and dimensionality reduction techniques in all of data science and engineering, particularly in image and signal processing, pattern extraction and recognition, machine learning, as well as other exploratory data analysis [69, 70]. It has been extensively used in various applications, for example, image noise estimation [71], denoising [72], video watermarking [73], video and image classification [74], analysis of functional magnetic resonance imaging data [75], and face recognition [76]. Here we will provide a brief review of the technique and the terminology used when discussing PCA.

Consider a dataset consisting of p-dimensional variables $y \in \mathbb{R}^p$ sampled from an unknown p-dimensional subspace, for such a dataset, PCA can be used to generate such an orthogonal projection onto a lower dimensional subspace, typically named principal subspace [77], which can explain most of the variance (statistical information) of the data. These orthogonal basis vectors spanning the principal subspace are called principal loading vectors. Similarly, the data projection onto these principal loading vectors are called principal components *(PC)*.

The notion, that is the basis of PCA based dimensionality reduction is that the first few principal components tend to capture most of the variability of the data. Thus, unless the data is sampled from a homoscedastic multivariate distribution, the principal components spanning the subspace where the data variance is minimal can be discarded from the overall basis without much loss of information. Similar methodology can also be used for noise suppression [72], i.e., noise contamination dominates the signals coming from the subspace with low variance as compared to signals spanned by the principal

components explaining high variance. Thus, by removing those principal components which span low variance signal subspace, we can essentially suppress the overall noise level present in the data.

3.2 Principal Component Analysis

In this section, we briefly present Principal Component Analysis technique, which is followed by a detailed simulated example motivating the proposed Block-PCA algorithm.

Consider a signal matrix $\boldsymbol{Y} \in \mathbb{R}^{n \times p}$ of rank $q \leq min(n, p)$, with n observations and p variables. Let $\boldsymbol{y}^i$ denote the i^{th} row of $\boldsymbol{Y}$, $\boldsymbol{y}_i$ denote the i^{th} column of $\boldsymbol{Y}$, for $i=1, \ldots, n$ with zero mean, and let $\sum = \boldsymbol{Y}^\top \boldsymbol{Y} = cov(\boldsymbol{y}^i)$ be the positive definite covariance matrix of size $p \times p$ estimated from the data itself. Eigen-value decomposition of the matrix $\sum$ (Gram matrix) is given by

$$\sum = \sum_{j=1}^{q} \lambda_j \boldsymbol{v}_j \boldsymbol{v}_j{}^\top \tag{3.1}$$

Where $\lambda_1 \geq \lambda_2 \geq \lambda_3 \geq \ldots \lambda_q > 0$ are the eigenvalues and $\boldsymbol{v}_i \in \mathbb{R}^p$ for $j = 1, \ldots, q$ are the corresponding eigenvectors of Σ. The dimensionality of the data can then be reduced by replacing the original data with $\boldsymbol{YV}$, where $\boldsymbol{V} = [\boldsymbol{v}_1, \boldsymbol{v}_2, \ldots, \boldsymbol{v}_q]$. The matrix $\boldsymbol{V}$ is an orthonormal matrix, and each column vector $\boldsymbol{Yv}_j$ is the respective principal component.

The vectors $\boldsymbol{v}_j$ are obtained by solving the following optimization problem:

$\boldsymbol{v}_j = argmax\ var(\boldsymbol{Yv})$ such that $\boldsymbol{v}_j{}^\top \boldsymbol{v}_j = 1$ and $\boldsymbol{v}_j{}^\top \boldsymbol{v}_k = 0$ for $k < j$ (3.2)

The vector v_j, here, is the j^{th} principal loading vector, the data projection $\boldsymbol{Y}\boldsymbol{v}_j$ is the j^{th} principal component, whereas the operator var(.) computes the variance. Each principal component $\boldsymbol{Y}\boldsymbol{v}_j$ captures $\lambda_j/(\sum_{j=1}^{q}\lambda_j) \times 100$ percent of the total variance. The strength of PCA lies in its ability to provide a relatively simple explanation of the underlying data structure if a small number of PCs ($q \ll p$) can be used to explain most of the data variance. A simple, yet effective, way to computing PCA is to use the singular value decomposition (SVD) method, i.e., the SVD of data matrix $\boldsymbol{Y}$ can be decomposed into the factor matrices given by

$$\boldsymbol{Y} = \boldsymbol{U}\Delta\boldsymbol{V}^{\top} \tag{3.3}$$

In the above decomposition, matrix $\boldsymbol{V}$ contains the right singular vectors (PC loadings), $\boldsymbol{U}$ contains left singular vectors, and Δ is a diagonal matrix with ordered singular values $\delta_1 \geq \delta_2 \geq \ldots, \geq \delta_q > 0$. Similarly, the matrix $\boldsymbol{U}\Delta$ are the PCs. Moreover, $\boldsymbol{U}$ and $\boldsymbol{V}$ are unitary, i.e., $\boldsymbol{U}^{\top}\boldsymbol{U} = \boldsymbol{V}^{\top}\boldsymbol{V} = \boldsymbol{I}_q$. The SVD of a matrix $\boldsymbol{Y}$ provides its closest rank-q matrix approximation $\widehat{\boldsymbol{Y}}_q$, where the closeness between $\boldsymbol{Y}$ and $\widehat{\boldsymbol{Y}}_q$ is quantified by the squared Frobenius norm of their difference, i.e., $\left\|\boldsymbol{Y} - \widehat{\boldsymbol{Y}}_q\right\|^2{}_F$.

3.3 Motivation

In this section, we discuss our motivation behind using the PCA for noise suppression. The idea becomes very intuitive if we could visualize the learned principal components, the singular values, and their respective PC loadings. Thus, to do this, let $\boldsymbol{s}(n)$ be a single channel clean time-domain speech signal, $\boldsymbol{y}(n)$ be the noise contaminated signal

with a specific signal to noise ratio, and $\boldsymbol{r}(n)$ be the corresponding noisy signal. Further assume that **S**(k,m), **Y**(k,m), and **R**(k,m) are their respective short-time Fourier transforms (STFT) matrices with size 1025×337. The magnitude of these STFTs is shown in Fig. 3.1. To generate these STFTs, we used Hamming window of size 1024, signal overlap of 64 samples (to obtain a dense matrix for visualization), and taking fast Fourier transform of size 2048. The sampling rate of the time-domain signal was $Fs = 8000$ samples/sec, leading to $\Delta f \approx 4\,Hz$. The signal to noise ratio was kept at 0 dB. The speech signal contained the voice of a male speaker saying, "The birch canoe slid on the smooth planks", acquired from the NOIZEUS corpus [78]. The noise (interference) signal, used here, was "babble" noise from the NOIZEUS corpus as well. We show these STFTs in Fig. 3.1.

By inspecting STFT of the clean signal (shown in Fig. 3.1 (a)) we can infer that most of the signal energy lies between 300 - 600 Hz range and silence blocks are clearly visible as well. On the other hand, the noise energy (shown in Fig. 3.1 (c)) has a good spread over the range of 300 - 3500 Hz, with significant energy between 300 - 600 Hz range as well. Moreover, its presence is consistent over the entire duration of the signal. The STFT corresponding to 0 dB SNR is given in Fig. 3.1(b).

Now let $\boldsymbol{X}$ be the magnitude of a STFT of interest ($\boldsymbol{S}, or\ \boldsymbol{Y}$), we take the SVD of $\boldsymbol{X}$ according to (4.3). The resulting singular values are diag(Δ), principal components are $\boldsymbol{U}\Delta$, and PC loadings are $\boldsymbol{V}$. To visualize the spread of variance over various PCs, we show the first 200 (out of 337) singular values in Fig. 3.2. From the slope of both curves, we can see that most of the data variance is captured by the few initial PCs. For noise free case, 20 - 40, and for noisy case, 60 - 80 PCs are capturing most of the variance. Moreover, higher

PCs (20 - onwards) are more affected by the noise as compared to the lower ones, thus, ideally, getting rid of such noisy components should lead to reduction in the overall noise level.

Additionally, for visualization, the initial 6 and 30 - 33 PCs and their respective loadings are also shown in Figs. 3.3 and 3.4 respectively. From Fig. 3.3 we can see that the initial 6 PCs learned from clean STFT, and noisy STFT show similar trends, showing less noise disturbance, whereas the noisy estimates of components 30 - 33, shown in Fig. 3.3, have clearly diverged from the clean ones. Fig. 3.4 shows similar trends with the PC loadings as well. Thus, to suppress noise, we can perform the signal approximation as

$$\hat{X} = \sum_{i=1}^{q} u_i \delta_i v_i^{\top} \tag{3.4}$$

where $\boldsymbol{u}_i \delta_i$ is the i^{th} PC, $\boldsymbol{v}_i$ is the i^{th} PC loading vector, and $q \ll p$. Here we would like to highlight two key issues with the framework, i.e., PCA performed over the entire signal STFT. Firstly, in the signal under consideration, the noise component is present for the entirety signal duration (see Fig. 3.1 (c)), whereas, signal of interest, the speech signal contains periods of silence corresponding to almost 20 - 30 % of the entire signal duration (see Fig. 3.1 (a)). Therefore, we can expect that the PCA computation might get biased towards learning such PCs and PC loadings, which can explain the entire duration of the signal. This is indeed the case as this effect is clearly visible in the PC 1 Fig. 3.3, and PC loading 1 Fig. 3.4. The first PC loading from the clean STFT (blue) clearly shows that the corresponding PC component has been able to capture the signal variance during the time

periods of voice activity. This, however, is not the case with the first components learned from noisy STFT, where these estimated components have ended up capturing significant noise contribution as well.

Secondly, if at any time during the signal recording, the energy of a few noisy frequencies gets higher than usual, then the PC and PC loading entries corresponding to these frequencies will become large i.e., the corresponding singular value will get large. As a result, these, otherwise noisy components, might get picked up for the STFT approximation leading to reduction in the effectiveness of noise suppression.

To circumvent these issues, we propose to perform noise suppression using PCA on small subsets of STFT, instead of the complete STFT. The idea is to look at a small number of time frames at a time and approximating them, one by one, instead of looking at the whole STFT. The advantage is that the SNR in each of these frames will be relatively higher than the SNR of the entire STFT, thus the very few initial PCs will be able to capture most of the data variance much effectively. Additionally, the noise increase in a specific window of time will be contained within that frame window and will not affect components in other adjacent frames. The details of this framework are provided in the next section.

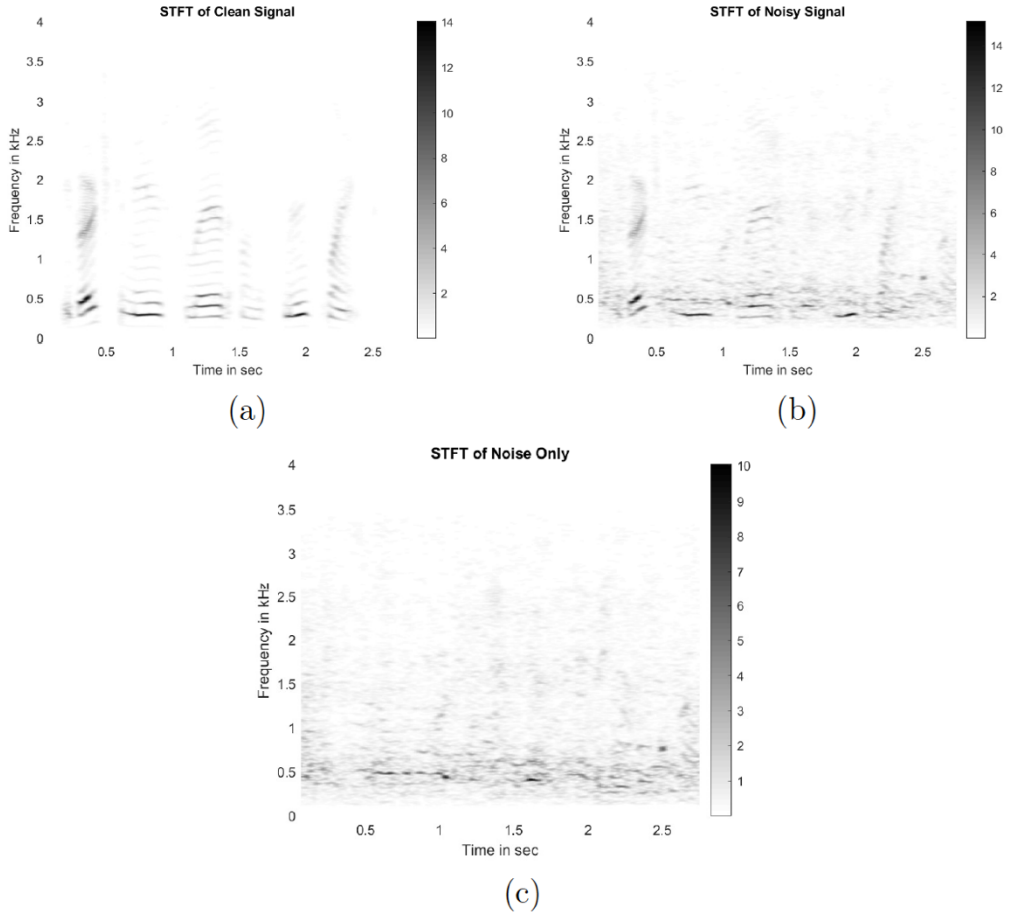


Figure 3.1: Magnitudes of Short-Time Fourier Transforms of (a) Clean Signal $s(n)$, (b) Noise Contaminated Signal $y(n)$ with 0 dB signal to noise ratio , and Noise $r(n)$ [80].

3.4 Block Principal Component Analysis

In this section, we formalize the block principal component analysis (termed Block-PCA) technique. Let $\boldsymbol{X} \in \mathbb{R}^{k \times m}$ be the STFT signal which we want to approximate with a low noise version $\widehat{\boldsymbol{X}}$. Here k is the number of frequency points, and m is the number of time frames. Let $b < m$ be a scalar denoting the block-size, $g = \lceil m/b \rceil$ denote the number of blocks, and a frame - block assignment vector $b = [1_b, 2_b, \ldots, g_b] \in \mathbb{R}^m$ here z_b denotes a vector of size b containing z. Let $\boldsymbol{X}_i$ be the matrix containing i^{th} block of $\boldsymbol{X}$, with indices coming from b.

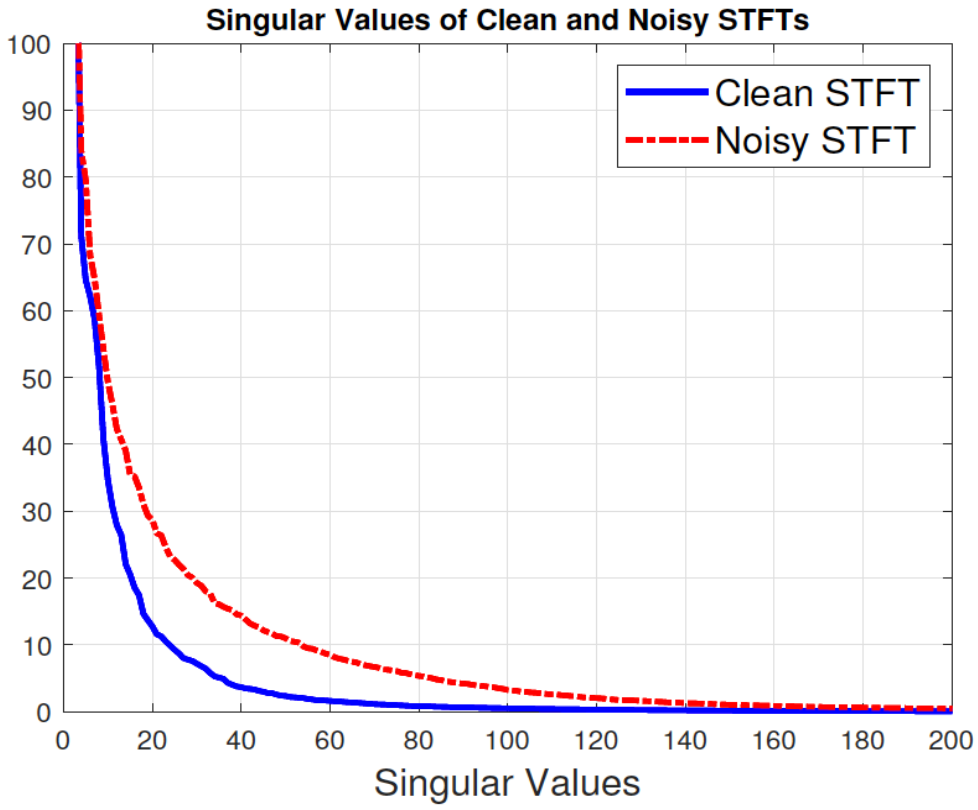


Figure 3.2: The initial 200 singular values corresponding to STFTs of Clean (in solid blue) and Noisy (in dash-dotted red) signals [80].

Then we approximate $\boldsymbol{X}_i$ by taking first taking its SVD, $\boldsymbol{U}\boldsymbol{\Delta}\boldsymbol{V}^{\top} = SVD(\boldsymbol{X}_i)$ and doing the following

$$\widehat{\boldsymbol{X}}_i = \sum_{j=1}^{q} \boldsymbol{u}_i \delta_i \boldsymbol{v}_i^{\top} \tag{3.5}$$

where $\boldsymbol{u}_i\delta_i$ is the j^{th} PC, $\boldsymbol{v}_i$ is the j^{th} PC loading vector, and $q \ll b$. Thus, in this approximation, we keep q most dominant components and discard the rest $(b - q)$. We do this for all $i = 1, 2, \ldots, g$ blocks, and recreate the complete approximation matrix $\widehat{\boldsymbol{X}} = [\widehat{\boldsymbol{X}}_1, \widehat{\boldsymbol{X}}_2, \ldots, \widehat{\boldsymbol{X}}_g]$ by concatenating all block approximations.

To visualize the effects of such approximation using Block-PCA, we applied our Block-PCA technique with block size $b = 70$ and $q = 5$ on the clean and noisy STFTs given in Fig. 3.1 (a,b), let's call the STFT under consideration $\boldsymbol{X}$ (taking on ($\boldsymbol{S}$ *or* $\boldsymbol{Y}$)). The singular values computed for each block $\boldsymbol{X}_i$ are shown in Fig.3.6 (a). Here we can see, that compared to Fig. 3.2, the singular values in each block drop much quickly, thus a few components can lead to a better approximation. Here we take $q = 5$ components per block to generate $\widehat{\boldsymbol{X}}_i s$. For completeness, these $q = 5$ singular values are also shown in Fig. 3.6 (b). The approximated complete $\widehat{\boldsymbol{X}}$ is shown in Fig. 3.5 (a), and the residual ($\widehat{\boldsymbol{X}} - \boldsymbol{X}$) is shown in Fig. 3.5 (b). By comparing the noisy STFT $\boldsymbol{X}$ (given in Fig. 3.5 (c)) and the approximated $\widehat{\boldsymbol{X}}$, we can see that the dominant features of our signal of interest (Fig. 3.1 (a)) are still present in the approximation, whereas most of the noise present in the 300 – 600 Hz range has been removed from the approximation and is delegated to the residual STFT.

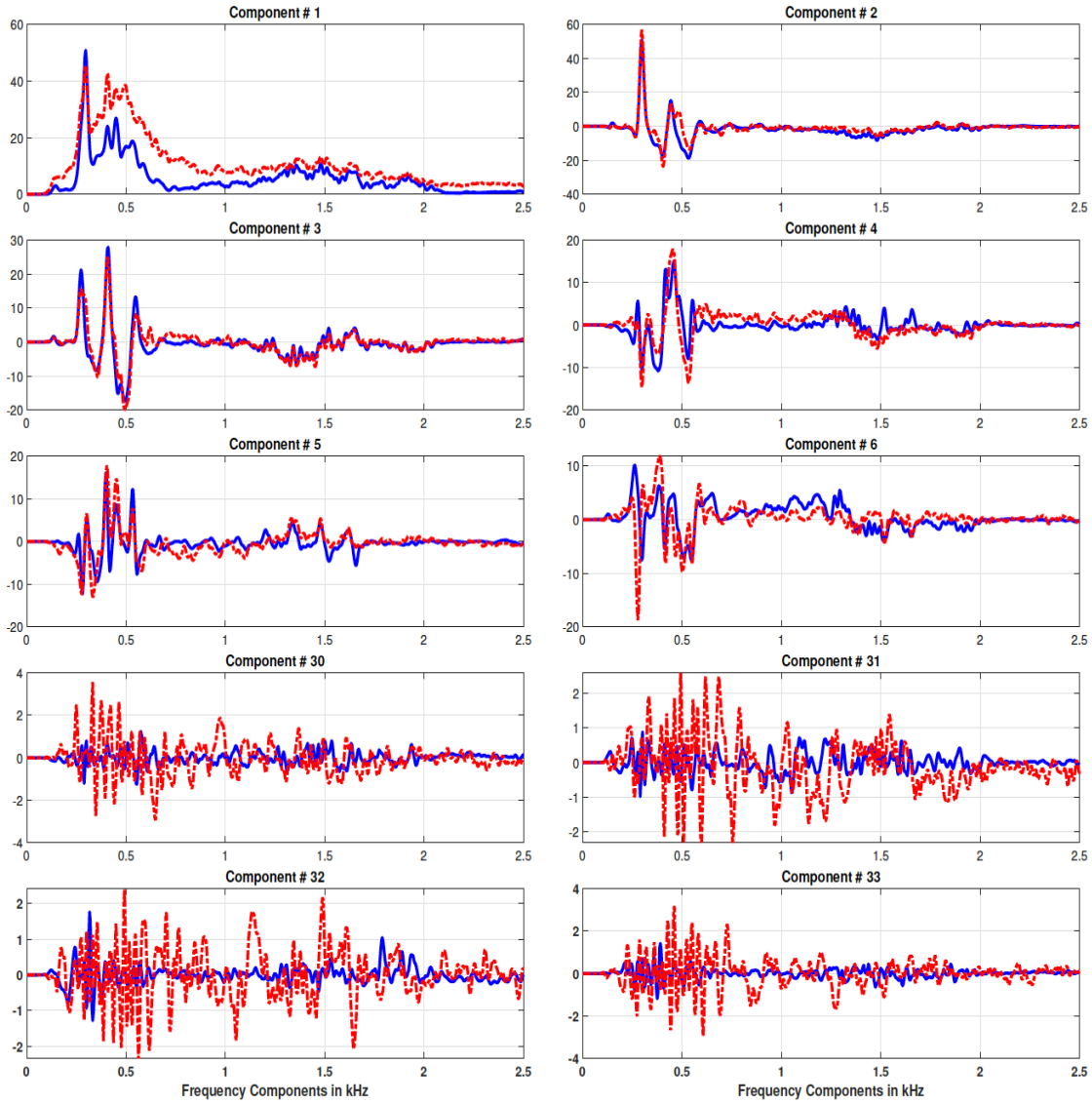


Figure 3.3: The Principal Components of STFTs of Clean (in solid blue) and Noisy (in dash-dotted red) signals [80].

For comparison, the STFT of noise only signal is also shown in Fig. 3.5 (d) which is very close to our residual STFT.

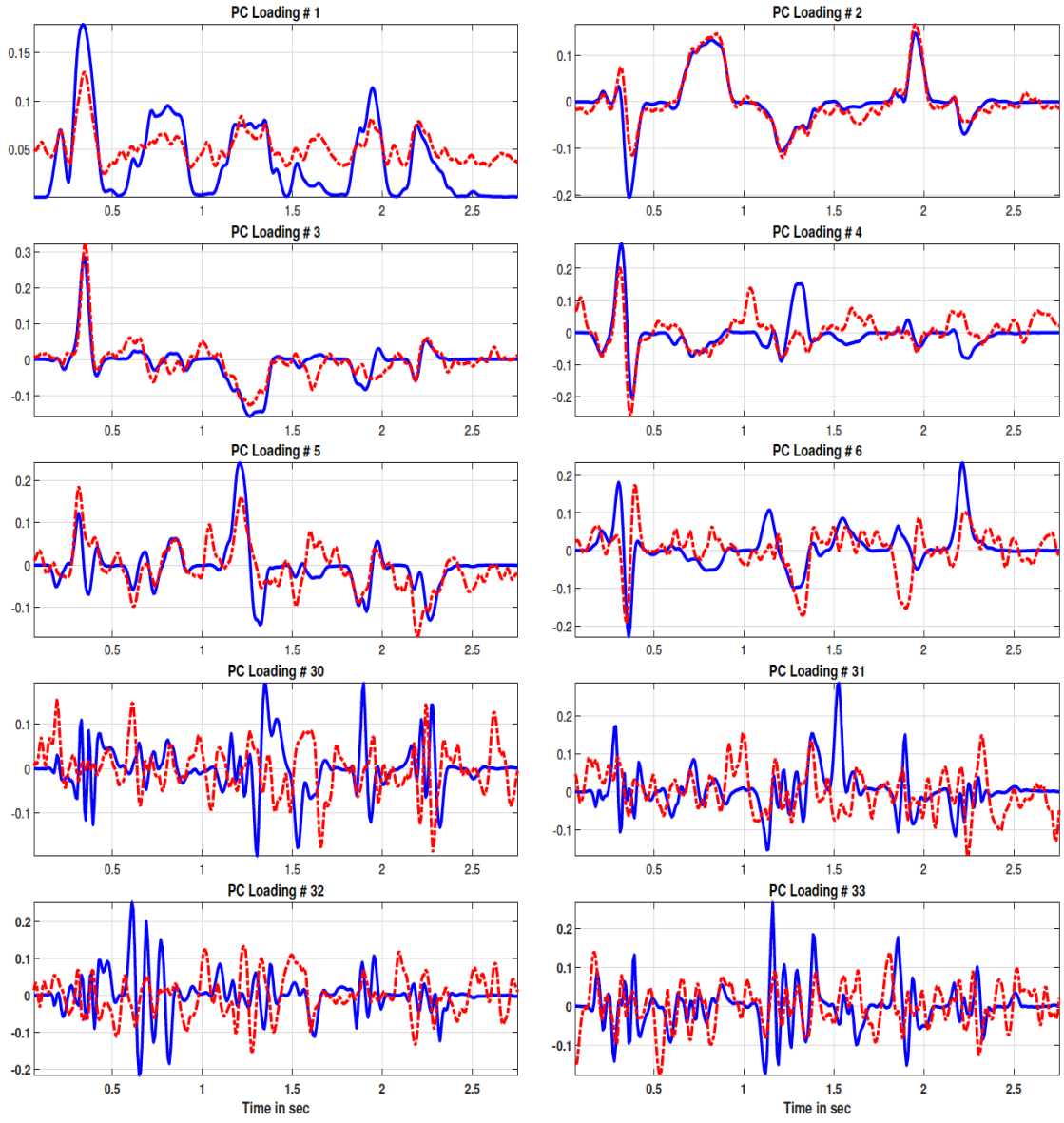


Figure 3.4: The PC Loadings of STFTs of Clean (in solid blue) and Noisy (in dash dotted red) signals [80].

The overall speech enhancement framework incorporating our noise suppression technique is outlined next.

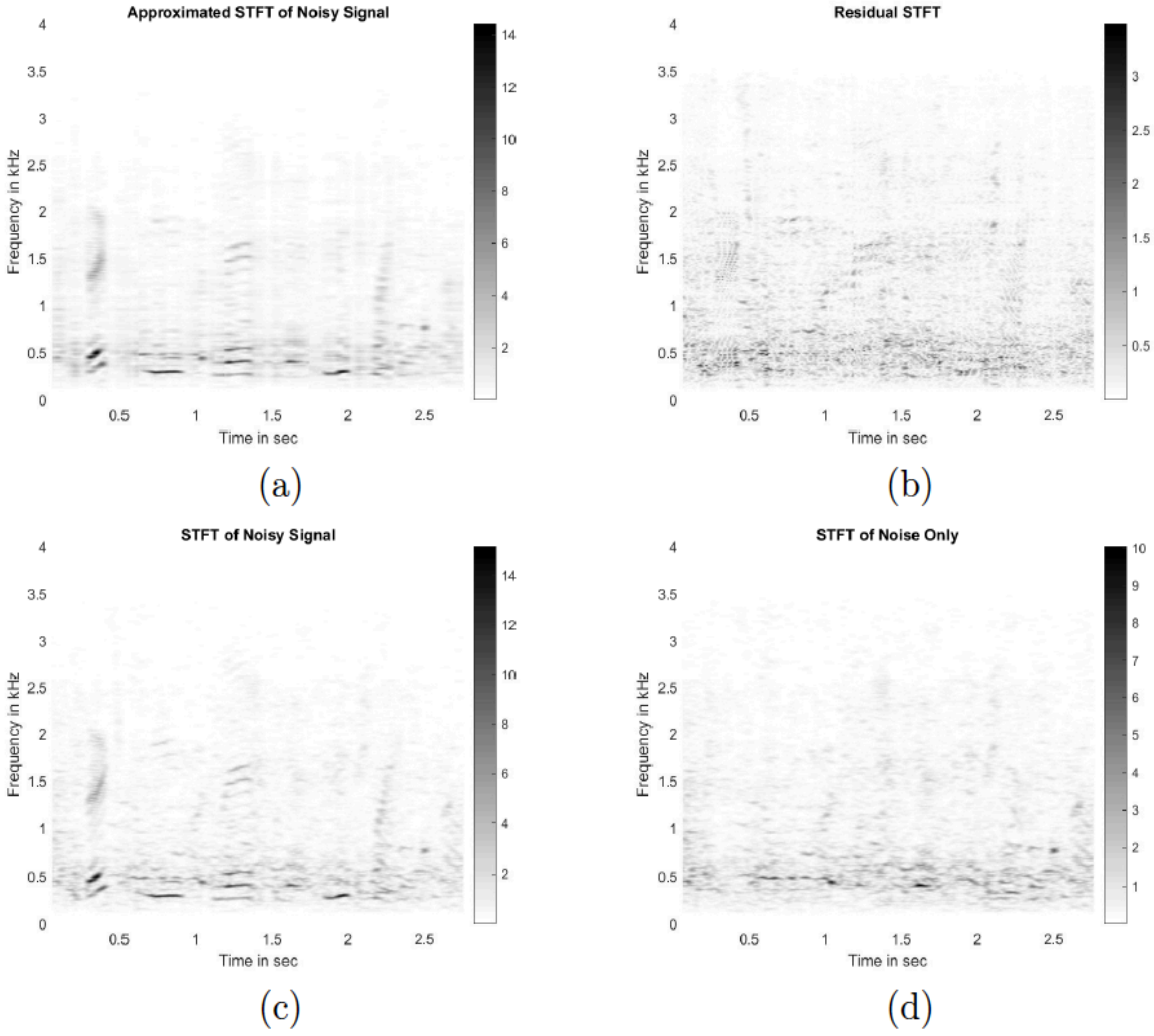


Figure 3.5: The approximated STFT $\widehat{\boldsymbol{X}}$ of noisy STFT (a) and the residual $(\widehat{\boldsymbol{X}} - \boldsymbol{X})$(b). For comparison, the noisy STFT $\boldsymbol{X} = \boldsymbol{Y}$ and noise only STFT from Fig. 3.1 are reproduced in (c) and (d) respectively [80].

3.5 Speech Enhancement Framework

In this section, we outline the steps performed to recover clean speech signal $\tilde{s}(n)$ from the noisy one $y(n)$.

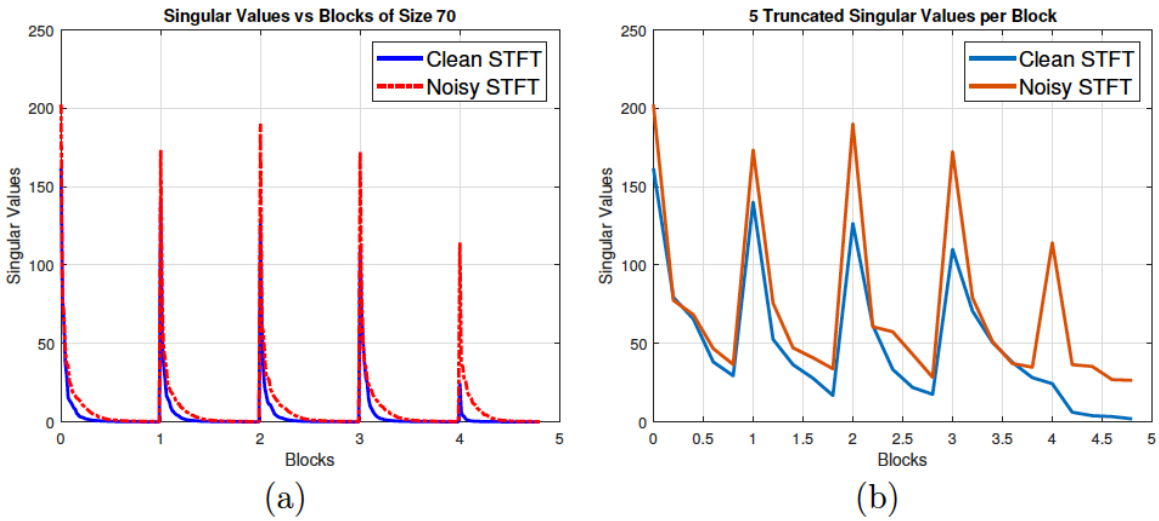


Figure 3.6: (a) Singular values corresponding to every block, and (b) the truncated singular values per block used for the approximated STFT $\hat{X}$ shown in Fig. 3.5 (a) [80].

1. Let y(n) denote the time-domain noisy signal vector, s(n) be the clean speech signal, and r(n) represent noise. All signals at this step are real-valued and are related by:

$$\boldsymbol{y}(n) = \boldsymbol{s}(n) + \boldsymbol{r}(n) \tag{3.6}$$

2. Using Hamming window of appropriate size, segment y(n) vector into multiple frames with appropriate level of overlap. The resulting segmented noisy signal is given by

$$\boldsymbol{y}(n,m) = \boldsymbol{s}(n,m) + \boldsymbol{r}(n,m) \tag{3.7}$$

where n represent the time points, and m represent the segment/frame number.

3. Take Fourier transform of the segmented noisy time-domain signal to get

$$\boldsymbol{Y}(k,m) = \boldsymbol{S}(k,m) + \boldsymbol{R}(k,m) \tag{3.8}$$

here $\boldsymbol{Y}(k,m)$ is the complex-valued truncated noisy spectrum with $k = n = 2 + 1$. Save $\boldsymbol{Y_\theta}(k,m) \triangleq \angle \boldsymbol{Y}(k,m)$, phase of the noisy spectrum,

4. Perform noise suppression on $|\boldsymbol{Y}(k,m)|$ via Block-PCA with block size b and retaining q components per block according to the steps provided in section 3.4, to get

$$\widehat{\boldsymbol{Y}}(k,m) = Block - PCA\ (|\boldsymbol{Y}(k,m)|, b, q) \tag{3.9}$$

here the resulting $\widehat{\boldsymbol{Y}}(k,m)$ is the real-valued magnitude of the approximated STFT.

5. Apply speech enhancement algorithm to the noise suppressed magnitude spectra $\widehat{\boldsymbol{Y}}(k,m)$ to obtain enhanced magnitude spectra $\left|\widehat{\boldsymbol{S}}(k,m)\right|$.

6. Generate the complex-valued spectra by combining the enhanced magnitude spectra with phase of the noisy spectrum as

$$\widehat{\boldsymbol{S}}(k,m) = \left|\widehat{\boldsymbol{S}}(k,m)\right| e^{j\,\boldsymbol{Y_\theta}(k,m)} \tag{3.10}$$

7. Take inverse Fourier transform of $\widehat{\boldsymbol{S}}(k,m)$ to get $\hat{\boldsymbol{s}}(n,m)$ and apply the general overlap and add (OLA) method to obtain the enhanced time-domain signal $\hat{\boldsymbol{s}}(n)$.

Chapter Four: Results and Discussion

In this chapter, we present detailed investigation about the performance improvement of using our proposed pre-processing step in the speech enhancement process. We investigate the noise suppression performance of the proposed block Principal component analysis (Block-PCA) algorithm in the overall speech enhancement pipeline discussed in section 3.5, especially when the signal to noise ratio (SNR) is very low. The speech enhancement algorithms used here are the minimum mean square error - short-time spectral amplitude (MMSE-STSA) estimator [8], and the multi-band spectral subtraction (MBSS) method [21]. These algorithms are respectably fast and have been shown to perform very well on speech signals with medium to high SNR. These algorithms, however, see severe performance degradation under low SNR levels. Thus, our aim is to show that under low SNR levels, applying these methods on noise suppressed magnitude STFTs instead of the regular STFTs can lead to significant improvement. This is indeed the case as we will show next.

The dataset used for the experiments is the NOIZEUS [78] database which contains 30 different sentences spoken by 6 different speakers. Out of the 30 sentences, half are spoken by male speakers and the remaining half by female speakers. We used the 5 noise types available in the NOIZEUS database, namely "Airport", "Babble", "Car", "Exhibition", and "Restaurant", to contaminate the clean speech signals corresponding to 6 SNRs $\in [-9, -6, -3, 0, 3, 6]$ dB levels.

The metrics used to evaluate the enhanced speech quality are segmental SNR (SSNR) [79], and the perceptual evaluation of speech quality (PESQ) [62]. In addition, we have used the composite measures C_{sig}, C_{bak}, and C_{ovl} [62] to rate the speech distortion, noise distortion, and the overall quality respectively of the enhanced speech signal. These composite measures are the linear combinations of PESQ, log likelihood ratio (LLR) [63], and weighted-slope spectral (WSS) distance [61] scores.

In all the experiments discussed next, the sampling rate of the speech signals was fixed at $Fs = 8000$ Hz and we used Hamming window of size 200 (25 ms duration), with overlap of 80 samples (10 ms) to compute the short-time Fourier transform of the signal under analysis. The FFT size was set equal to the chosen window size. Let $\boldsymbol{s}(n)$ be the clean speech signal and $\boldsymbol{r}(n)$ be the noise signal of a specific type. We generate the noise contaminated signal $\boldsymbol{y}(n)$ with a specific SNR level by adding $\boldsymbol{r}(n)$ with specific energy to $\boldsymbol{s}(n)$. We perform speech enhancement using the framework outlined in section 3.5 with and without performing noise suppression via Block-PCA to analyze its effectiveness. We store the performance scores for the noisy input signal, MMSE-STSA only, MMSE-STSA with Block-PCA, MBSS only, and MBSS with Block-PCA. We repeat this process for all 30 speech signals, all 6 SNR levels, and for all noise types as discussed at the start of this section. The block size (b) and components to retain (q) are selected by performing a 2-D grid search over b = [40; 50; 60; 70] and q = [15, 16, . . ., 26] and choosing the combination which leads to highest performance. These results are presented and analyzed next.

4.1 PESQ and SSNR performance for Male and Female speakers under Babble noise contamination

As we know that the frequency characteristics of male and female speakers are different. Thus, to look at the performance separately we have presented two speech enhancement performance metrics PESQ and SSNR for male and female speakers separately in Table. 4.1. The scores given here, as well as tabulated later are all average results. Consider the PESQ scores for both male as well as female speakers under severe noise contamination of -9 and -6 dB SNR, we can see that the enhanced speech signal recovered by both MMSE-STSA and MBSS methods have scores below that of input noisy speech signal itself. This is expected as these methods have been shown to perform poorly under low SNR levels. However, when combined with noise suppressing Block-PCA algorithm, both methods recover speech signals with higher PESQ scores, highlighting that using the approximated STFT, after noise suppression has led to better performance as compared to when using the original STFTs. The PESQ gains with respect to the regular methods seen here are significant with 0.86 and 0.47 for -9 and -6 dB SNR respectively. The same trend can be seen over SSNR metric for -9 and -6 dB SNRs as well. These gains get smaller when the SNR levels increase, which was predictable. Moreover, the scores for MMSE-STSA + Block-PCA are still better than their competitors and are shown

Table 4.1: Average PESQ and SSNR scores for Male and Female speakers under Babble noise contamination over multiple SNR levels. Best results are highlighted in **BOLD** [80].

		PESQ			SSNR		
SNR dB	Method	Male	Female	Overall	Male	Female	Overall
-9	Noisy	1.376	1.231	1.303	-8.205	-8.253	-8.229
	MMSE-STSA	1.135	0.991	1.063	-5.609	-5.838	-5.724
	MMSE-STSA + BPCA	1.978	1.742	**1.860**	-5.084	-5.043	**-5.063**
	MBSS	1.342	1.015	1.178	-6.082	-6.144	-6.113
	MBSS + BPCA	1.728	1.568	1.648	-6.057	-6.053	-6.055
-6	Noisy	1.552	1.252	1.402	-7.103	-7.200	-7.152
	MMSE-STSA	1.297	1.307	1.302	-4.707	-4.83	-4.769
	MMSE-STSA + BPCA	1.832	1.723	**1.777**	-4.213	-4.261	**-4.237**
	MBSS	1.439	1.170	1.304	-5.180	-5.235	-5.208
	MBSS + BPCA	1.775	1.463	1.619	-5.090	-5.13	-5.110
-3	Noisy	1.695	1.379	1.537	-5.784	-5.913	-5.848
	MMSE-STSA	1.603	1.596	1.600	-3.776	-3.829	-3.802
	MMSE-STSA + BPCA	1.807	1.793	**1.800**	-3.418	-3.490	**-3.454**
	MBSS	1.675	1.499	1.587	-4.168	-4.209	-4.188
	MBSS + BPCA	1.822	1.636	1.729	-4.116	-4.132	-4.124
0	Noisy	1.855	1.630	1.742	-4.284	-4.42	-4.352
	MMSE-STSA	1.836	1.879	1.858	-2.807	-2.803	-2.805
	MMSE-STSA + BPCA	1.943	1.959	**1.951**	-2.531	-2.602	**-2.566**
	MBSS	1.852	1.742	1.797	-3.211	-3.280	-3.245
	MBSS + BPCA	1.943	1.838	1.890	-3.142	-3.269	-3.205
3	Noisy	2.011	1.838	1.925	-2.634	-2.767	-2.700
	MMSE-STSA	2.079	2.101	2.090	-1.802	-1.799	-1.800
	MMSE-STSA + BPCA	2.151	2.137	**2.144**	-1.583	-1.592	**-1.587**
	MBSS	2.033	1.998	2.015	-2.166	-2.215	-2.190
	MBSS + BPCA	2.118	2.045	2.081	-2.093	-2.247	-2.170
6	Noisy	2.167	2.043	2.105	-0.848	-0.976	-0.912
	MMSE-STSA	2.280	2.274	2.277	-o. 760	-0.888	-0.824
	MMSE-STSA + BPCA	2.350	2.312	**2.331**	-0.567	-0.693	**-0.630**
	MBSS	2.233	2.204	2.219	-1.262	-1.177	-1.219
	MBSS + BPCA	2.299	2.246	2.272	-1.058	-1.183	-1.120

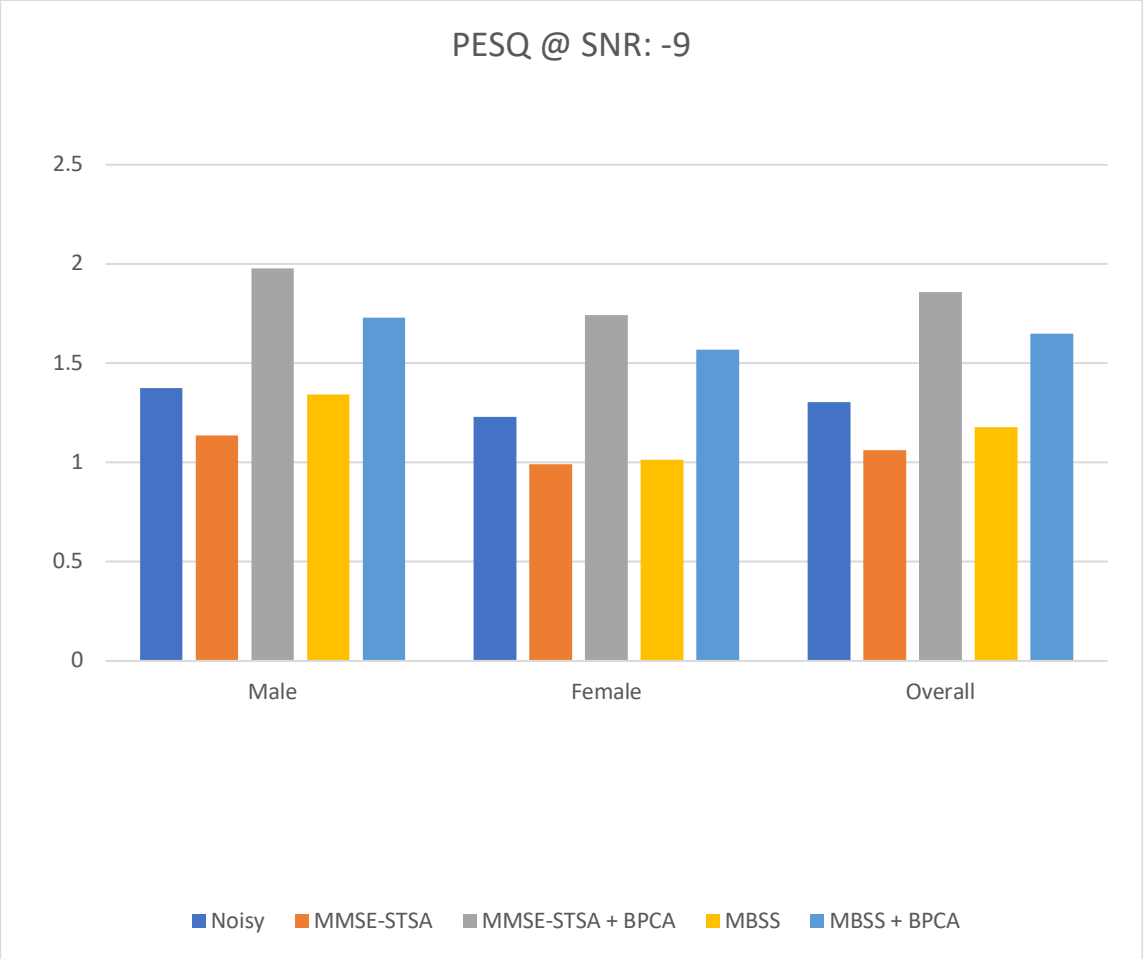


Figure 4.1.1: Average PESQ scores in -9 dB SNR for male and female speakers under 'Babble' noise contamination [80].

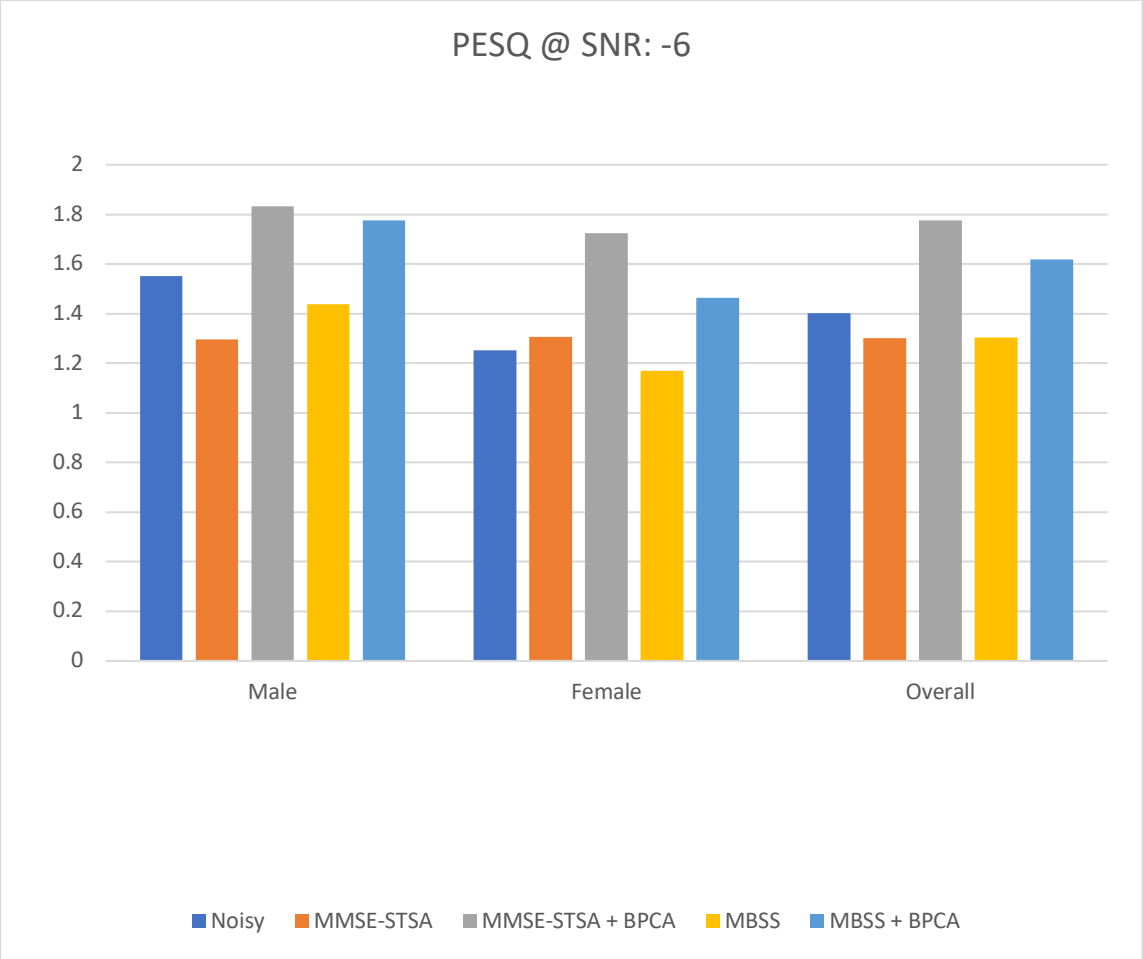


Figure 4.1.2: Average PESQ scores in -6 dB SNR for male and female speakers under 'Babble' noise contamination [80].

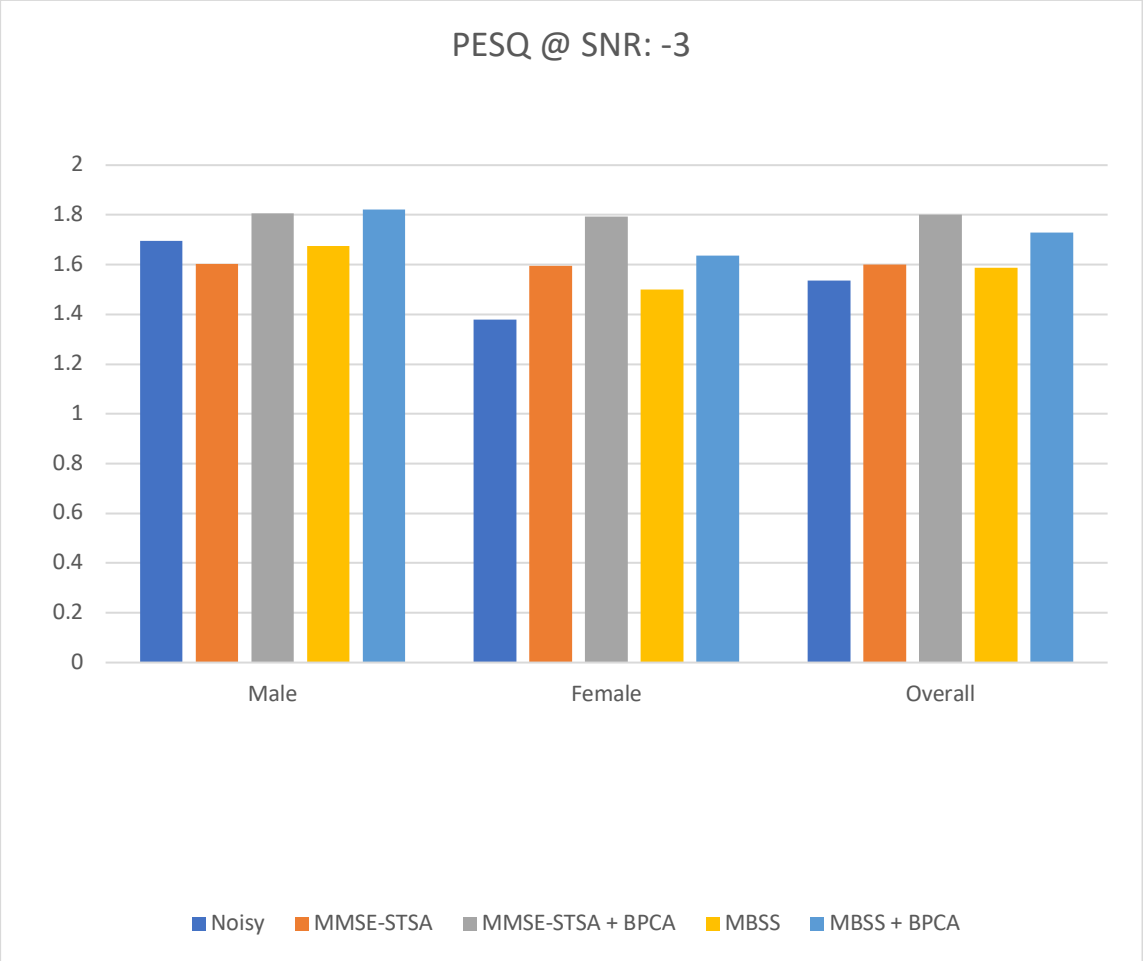


Figure 4.1.3: Average PESQ scores in -3 dB SNR for male and female speakers under 'Babble' noise contamination [80].

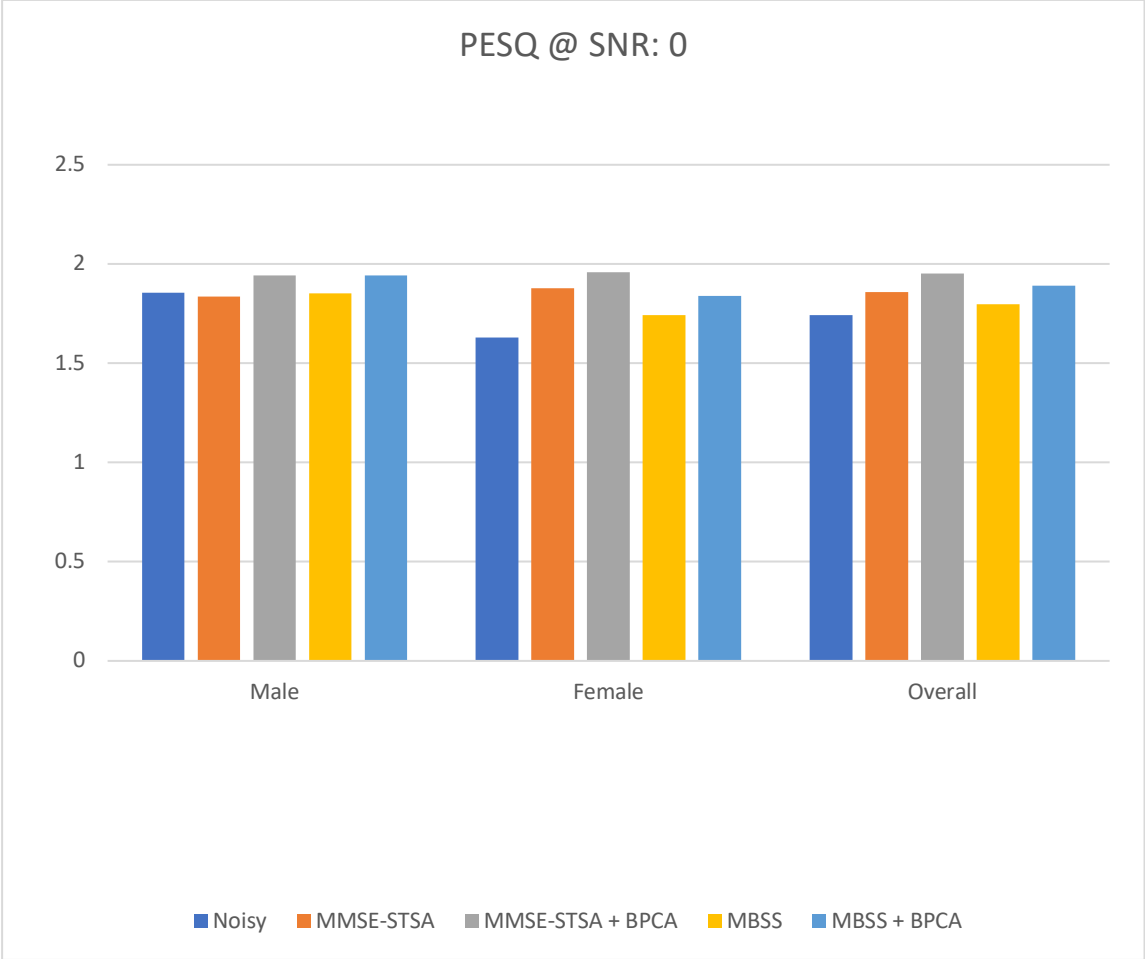


Figure 4.1.4: Average PESQ scores in 0 dB SNR for male and female speakers under 'Babble' noise contamination [80].

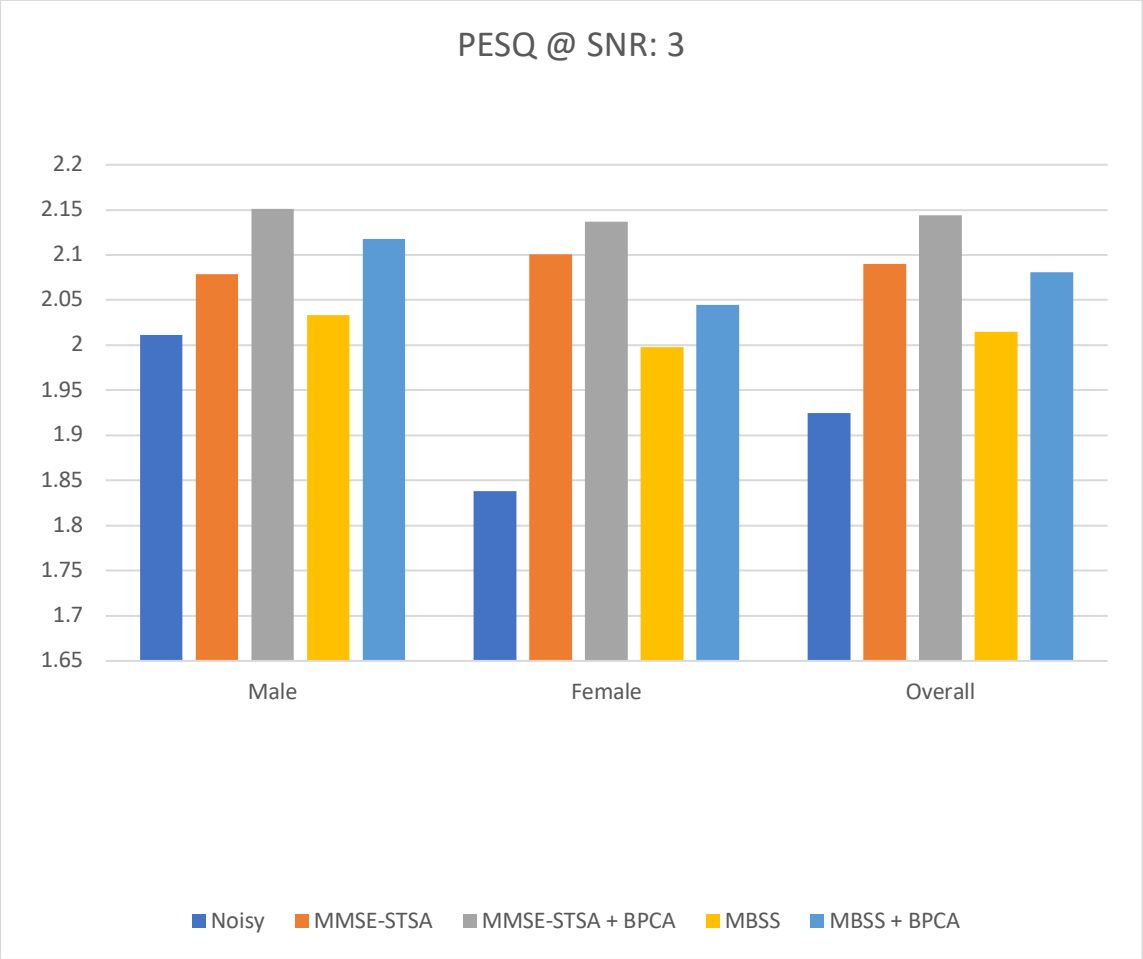


Figure 4.1.5: Average PESQ scores in 3 dB SNR for male and female speakers under 'Babble' noise contamination [80].

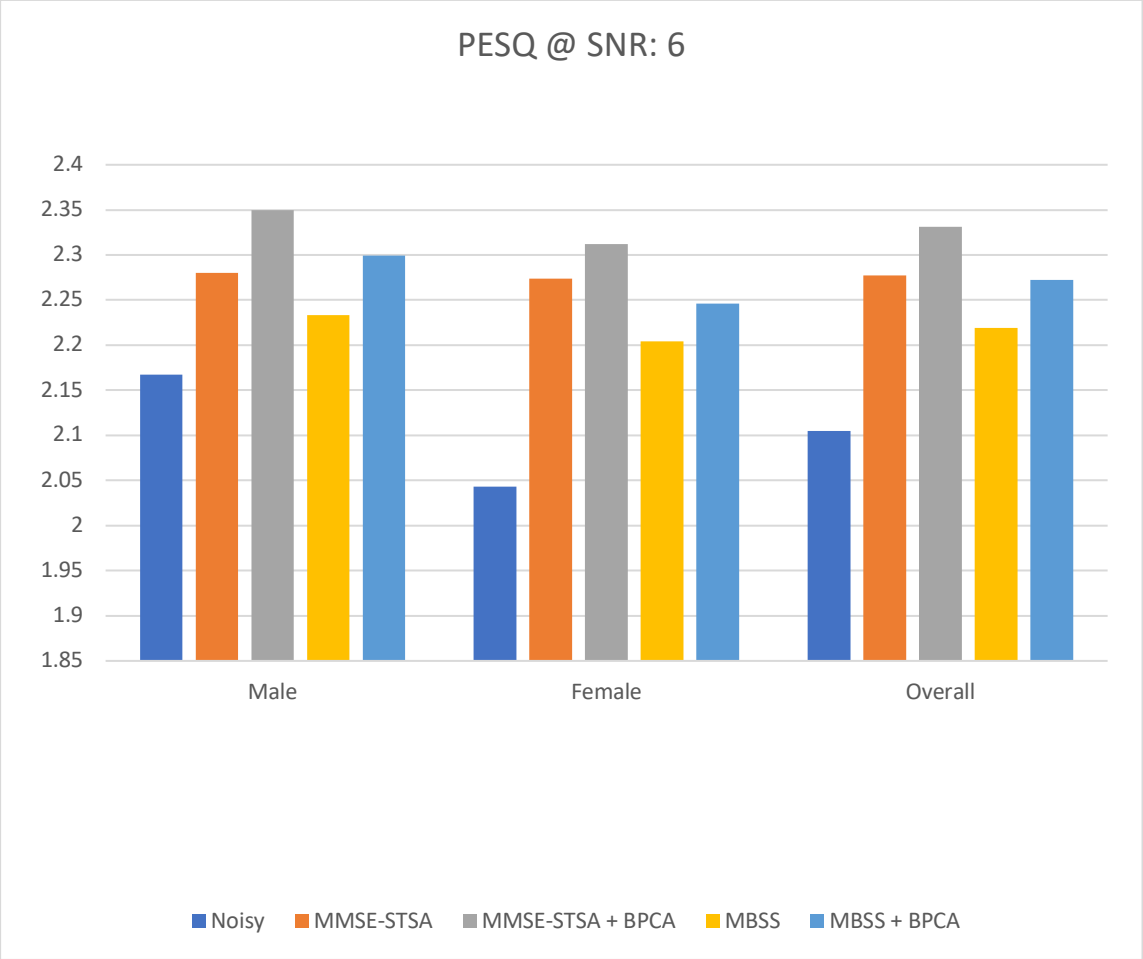


Figure 4.1.6: Average PESQ scores in 6 dB SNR for male and female speakers under 'Babble' noise contamination [80].

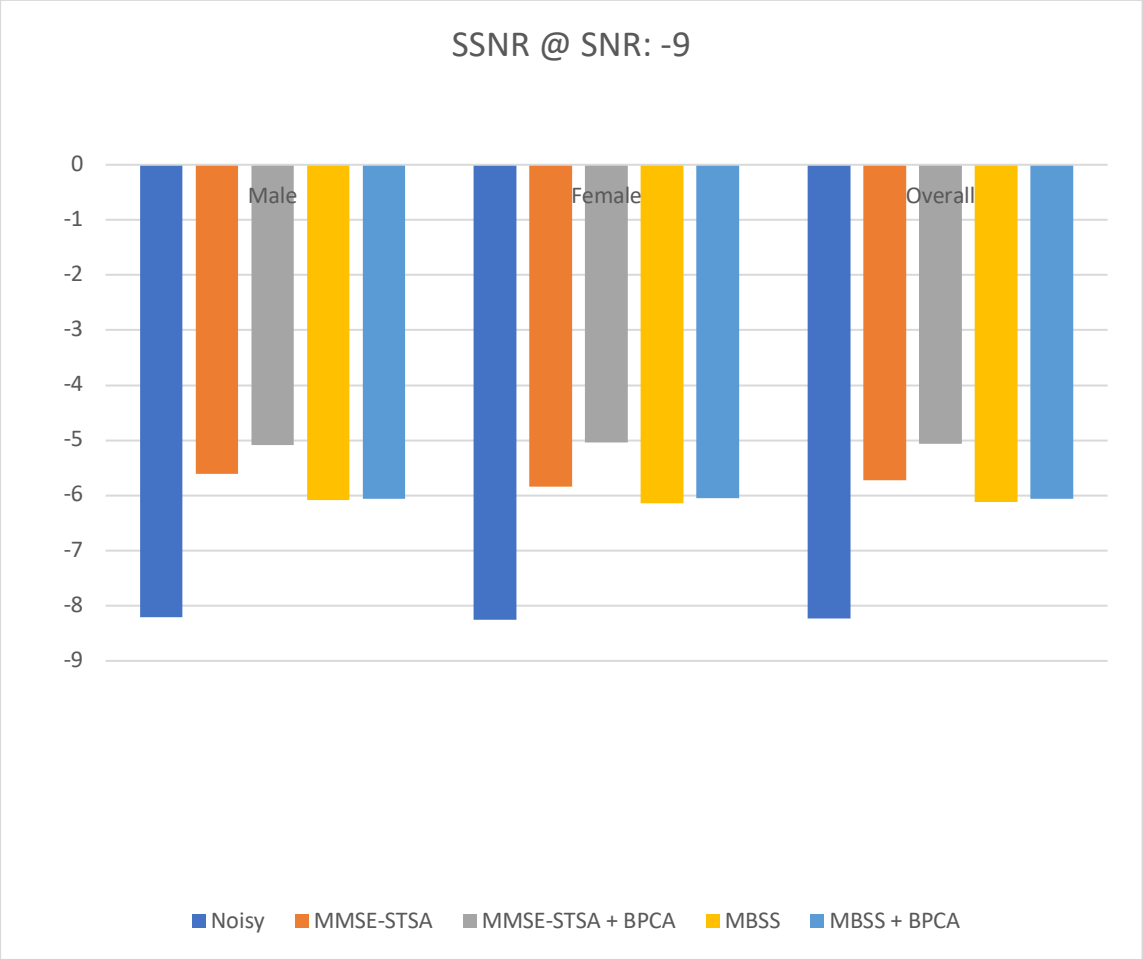


Figure 4.1.7: Average SSNR scores in -9 dB SNR for male and female speakers under 'Babble' noise contamination [80].

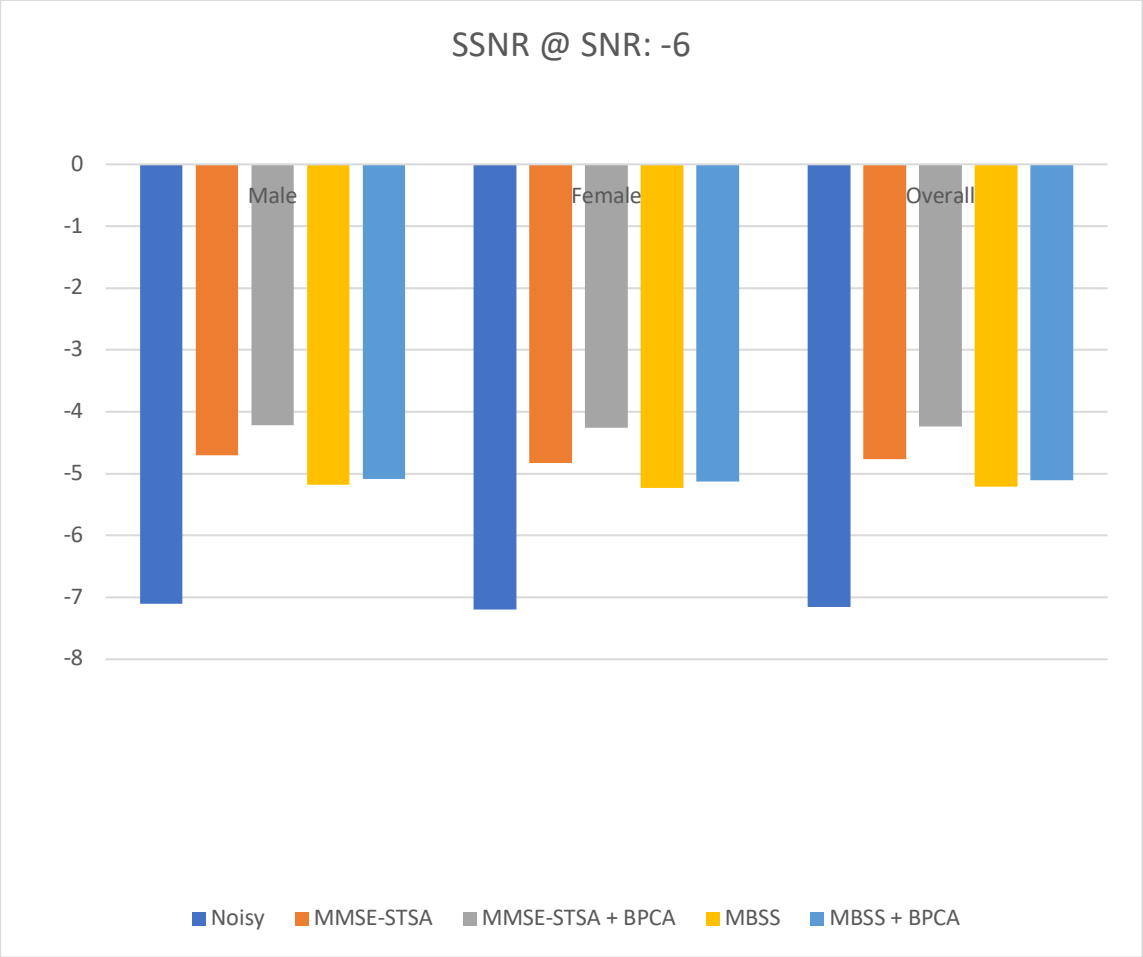


Figure 4.1.8: Average SSNR scores in -6 dB SNR for male and female speakers under 'Babble' noise contamination [80].

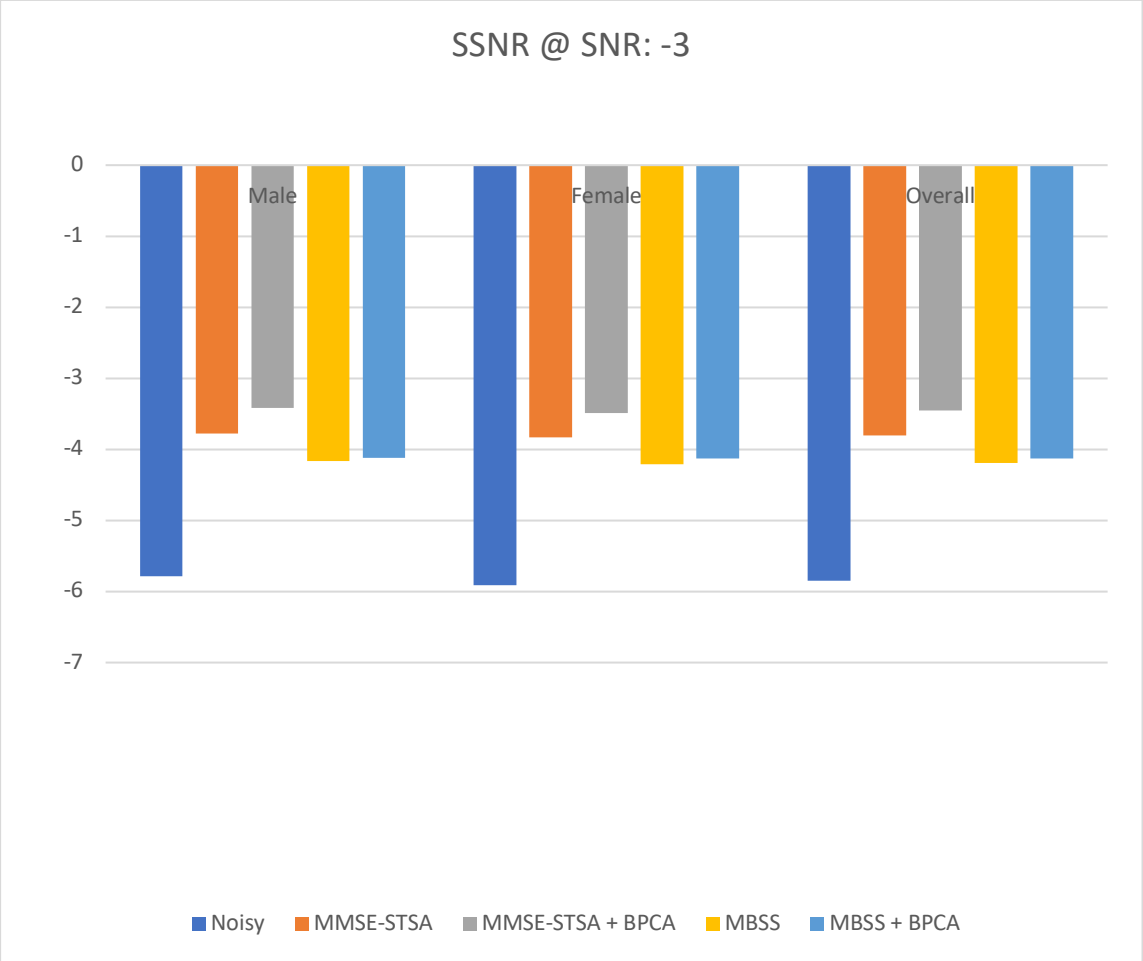


Figure 4.1.9: Average SSNR scores in -3 dB SNR for male and female speakers under 'Babble' noise contamination [80].

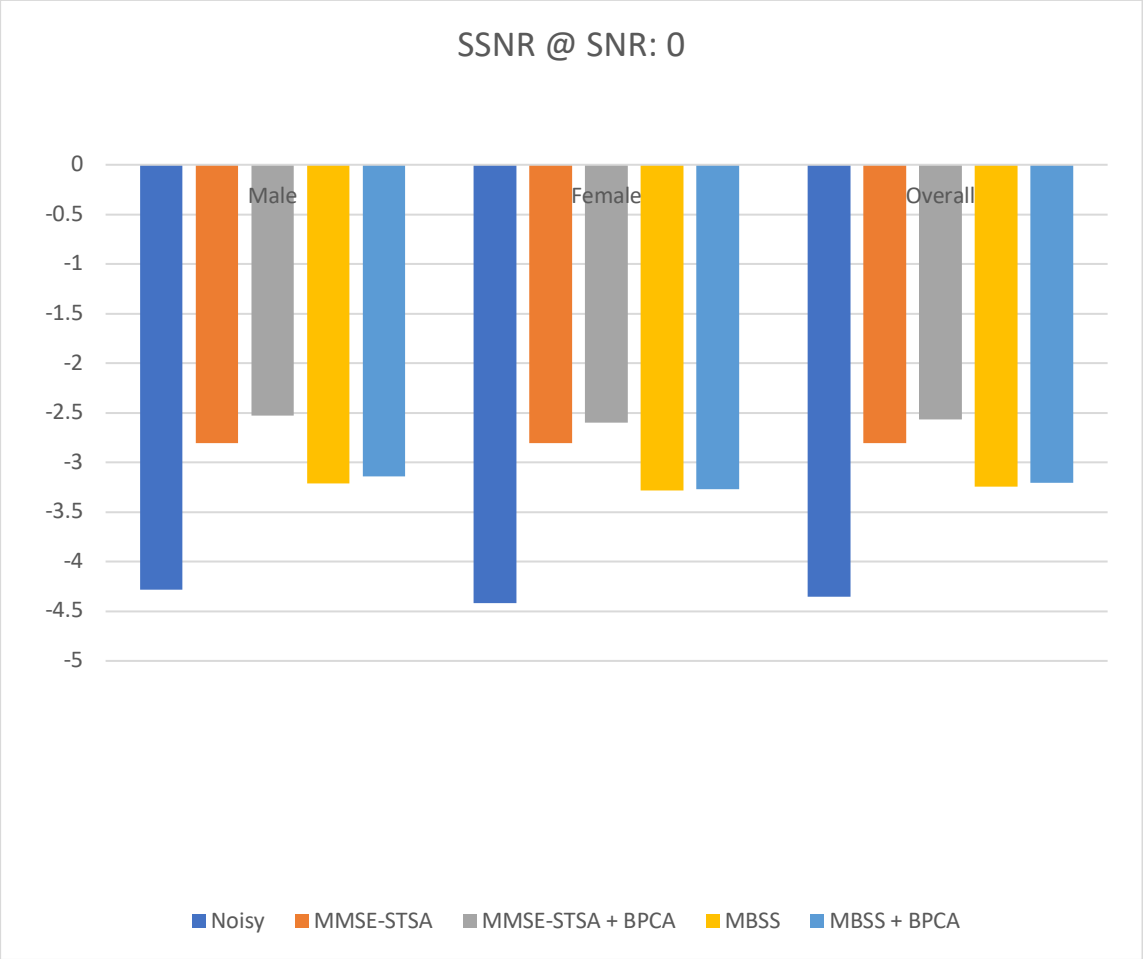


Figure 4.1.10: Average SSNR scores in 0 dB SNR for male and female speakers under 'Babble' noise contamination [80].

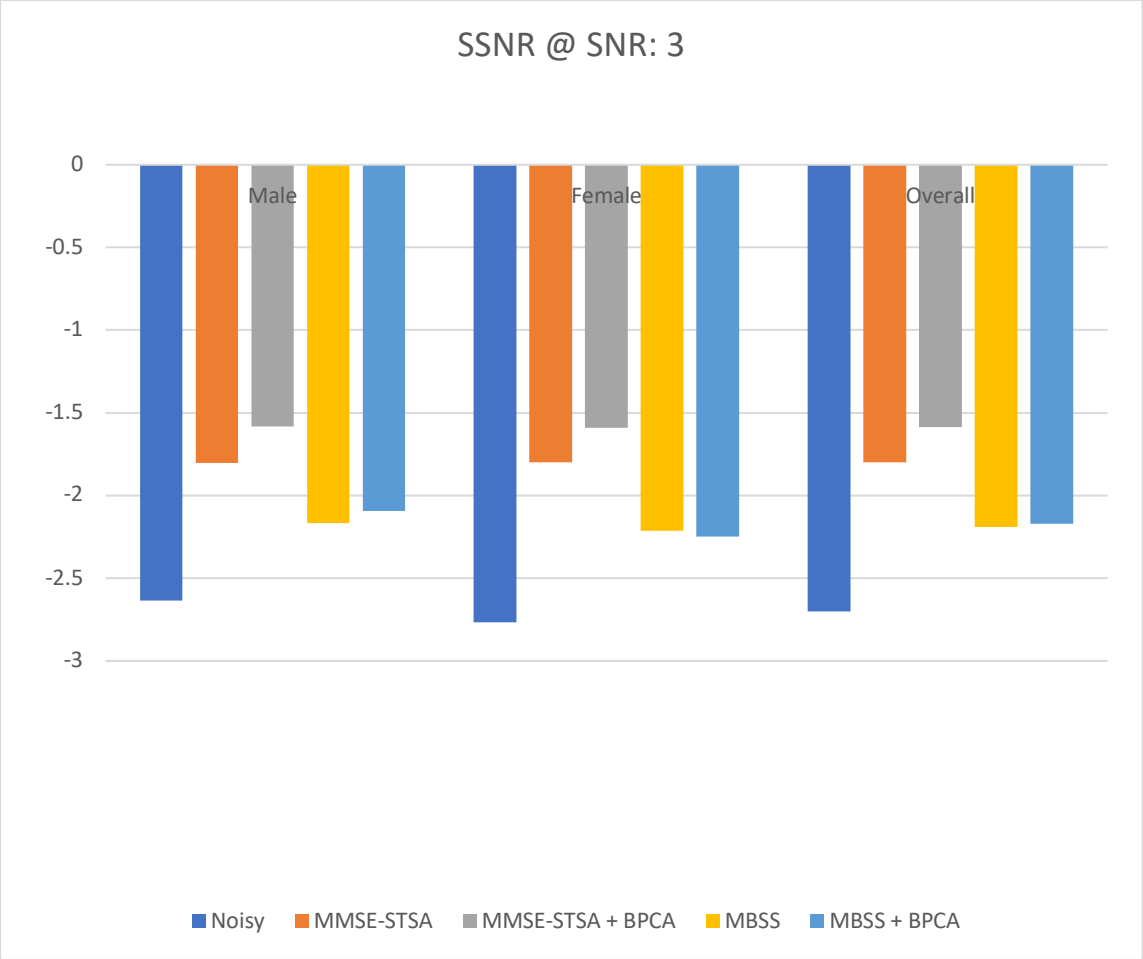


Figure 4.1.11: Average SSNR scores in 3 dB SNR for male and female speakers under 'Babble' noise contamination [80].

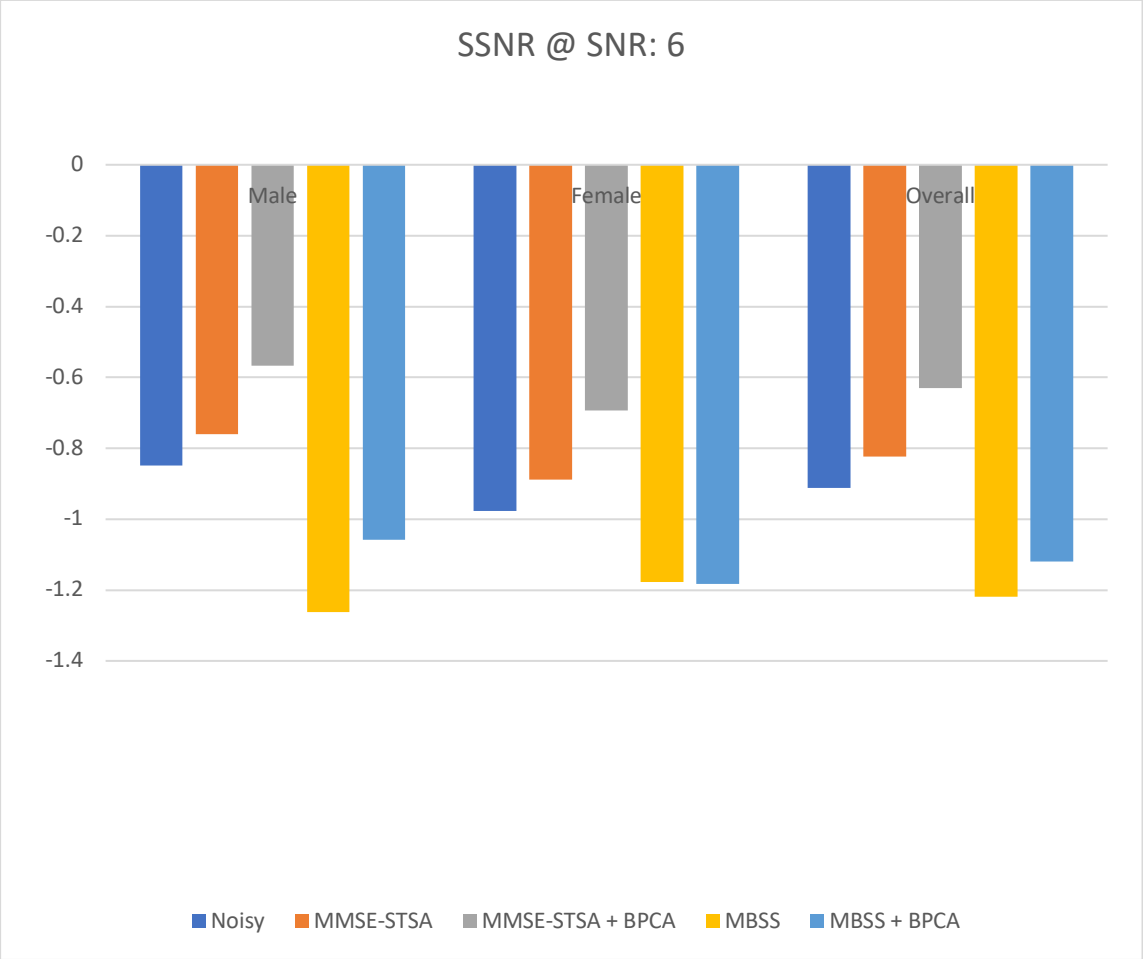


Figure 4.1.12: Average SSNR scores in 6 dB SNR for male and female speakers under 'Babble' noise contamination [80].

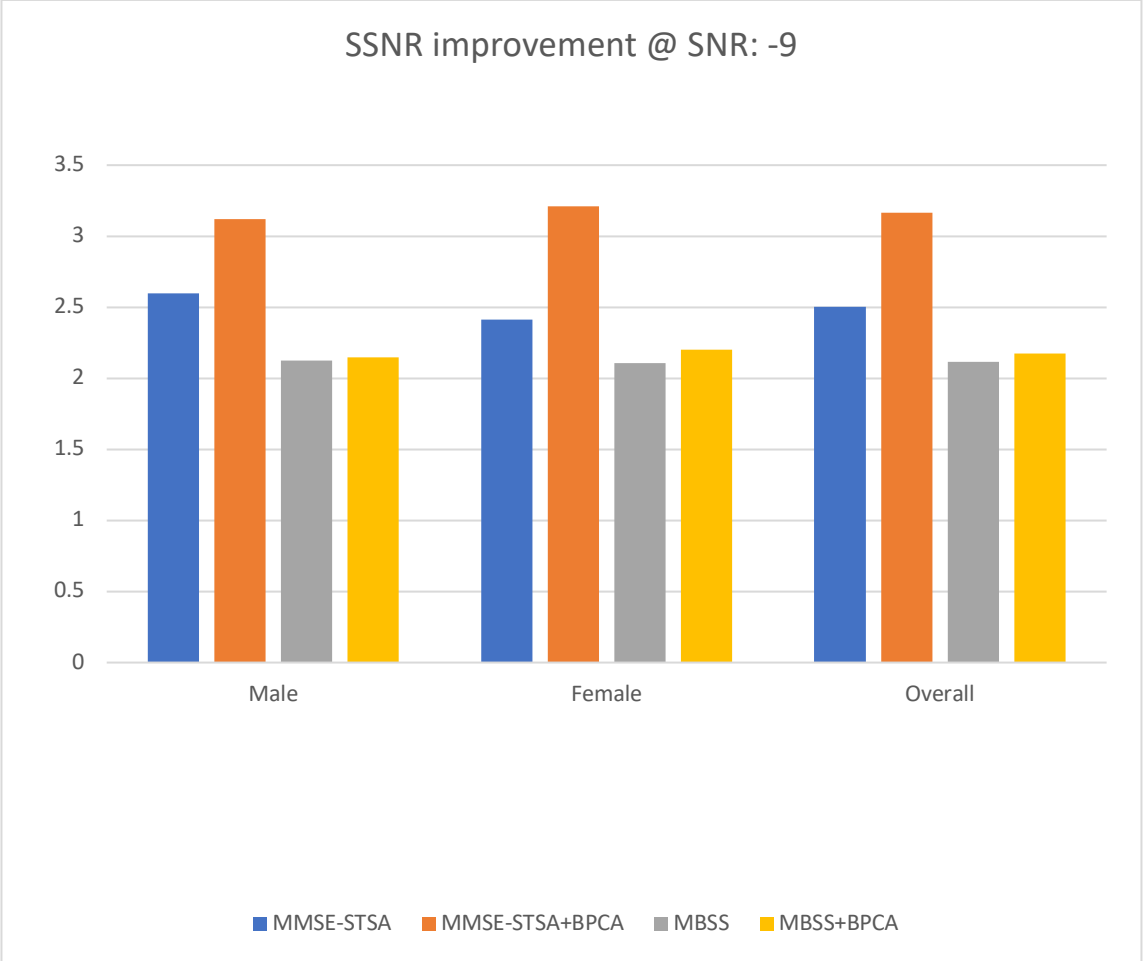


Figure 4.1.13: Average SSNR improvement in -9 dB with regard to the noise floor scores for male and female speakers under 'Babble' noise contamination [80].

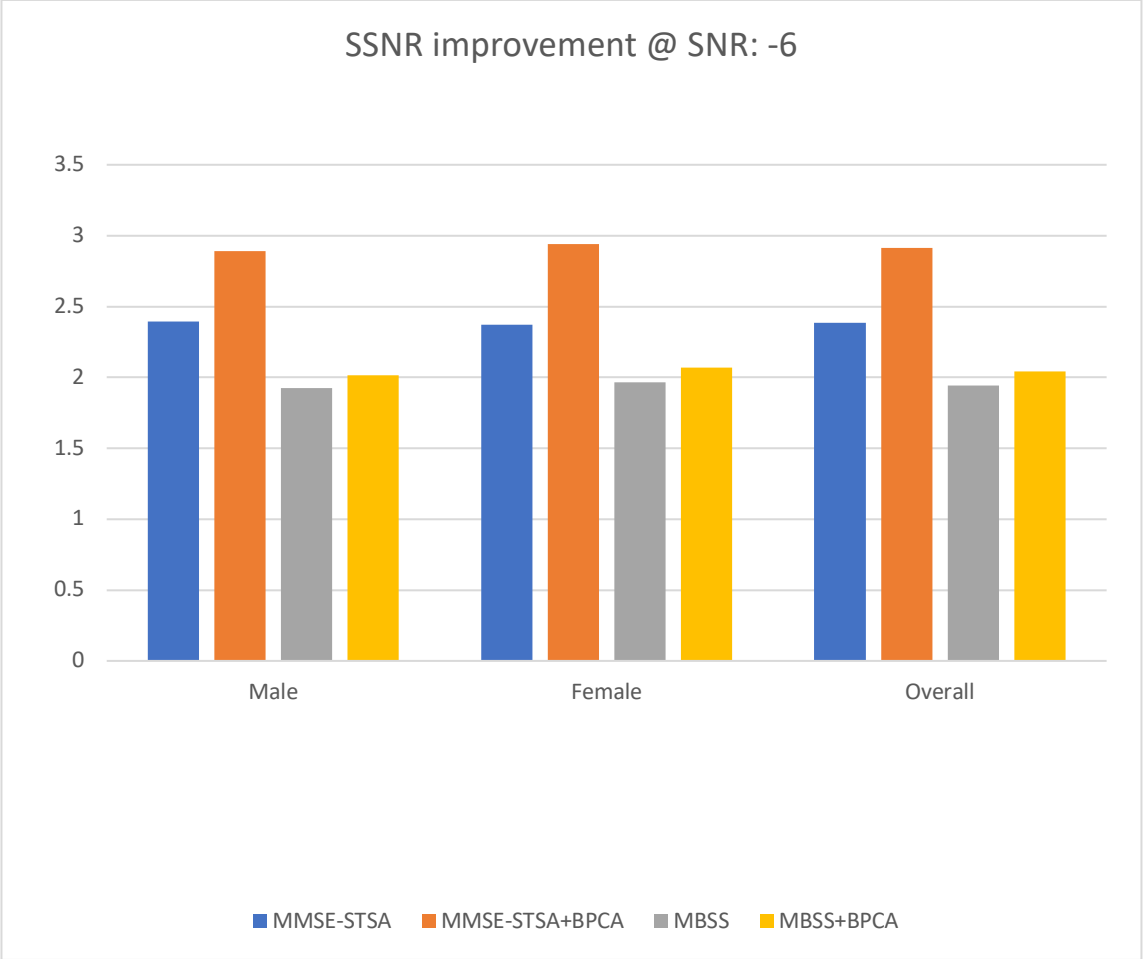


Figure 4.1.14: Average SSNR improvement in -6 dB with regard to the noise floor scores for male and female speakers under 'Babble' noise contamination [80].

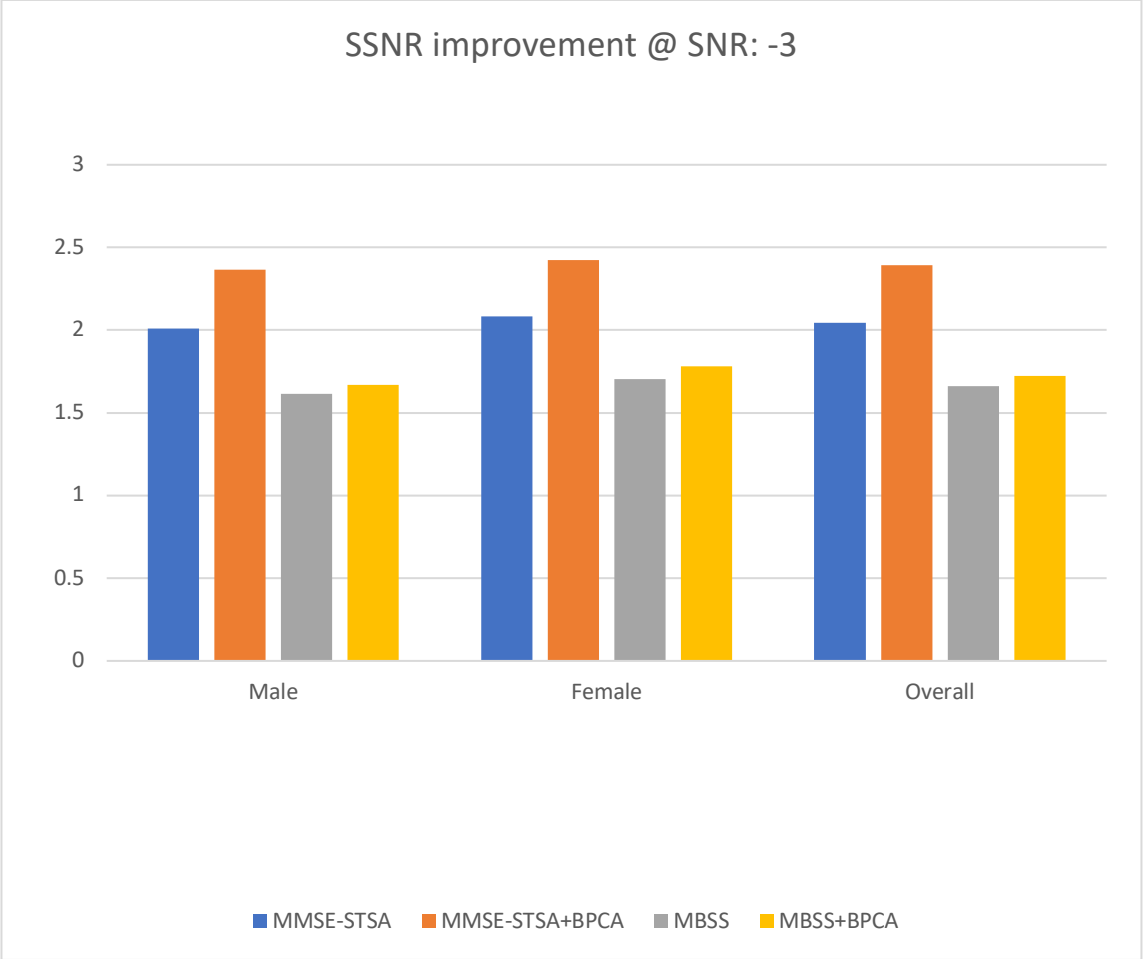


Figure 4.1.15: Average SSNR improvement in -3 dB with regard to the noise floor scores for male and female speakers under 'Babble' noise contamination [80].

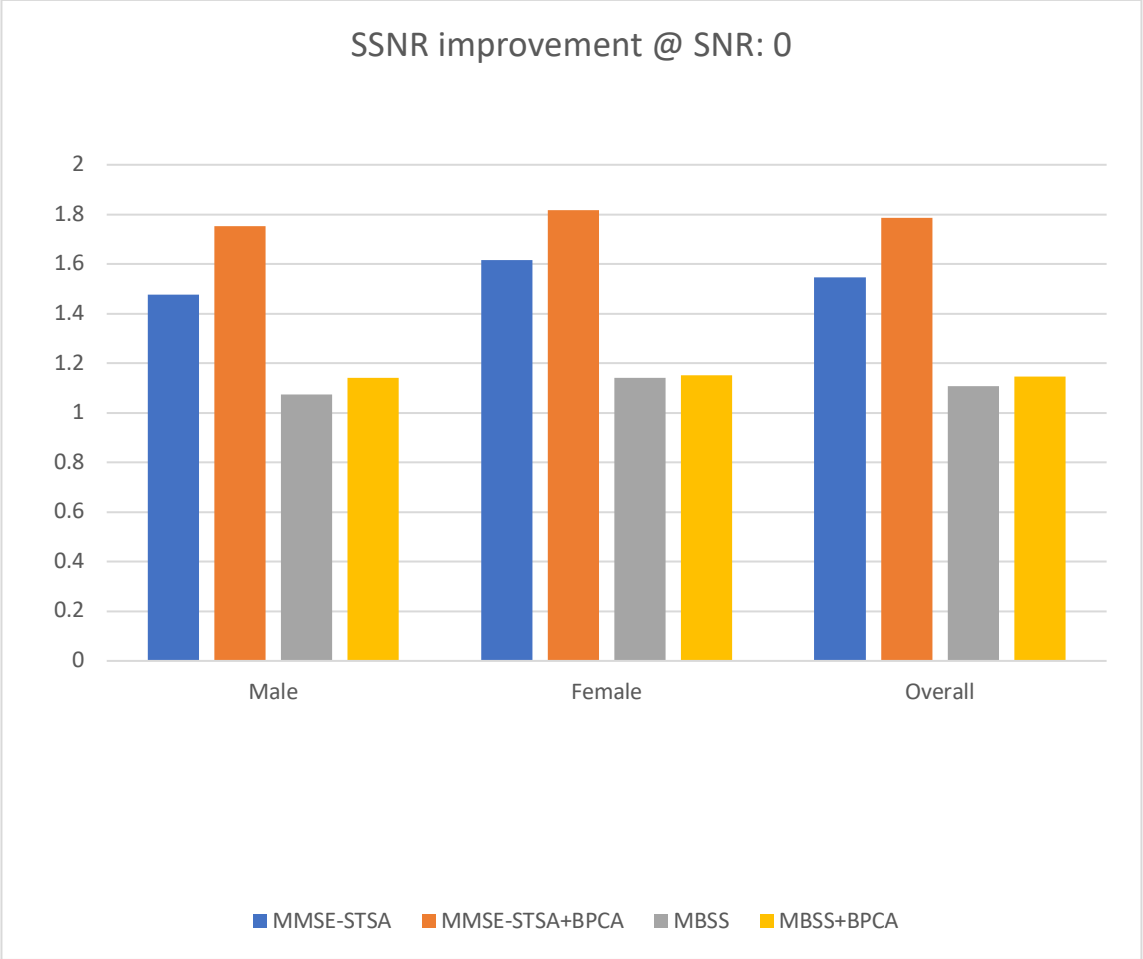


Figure 4.1.16: Average SSNR improvement in 0 dB with regard to the noise floor scores for male and female speakers under 'Babble' noise contamination [80].

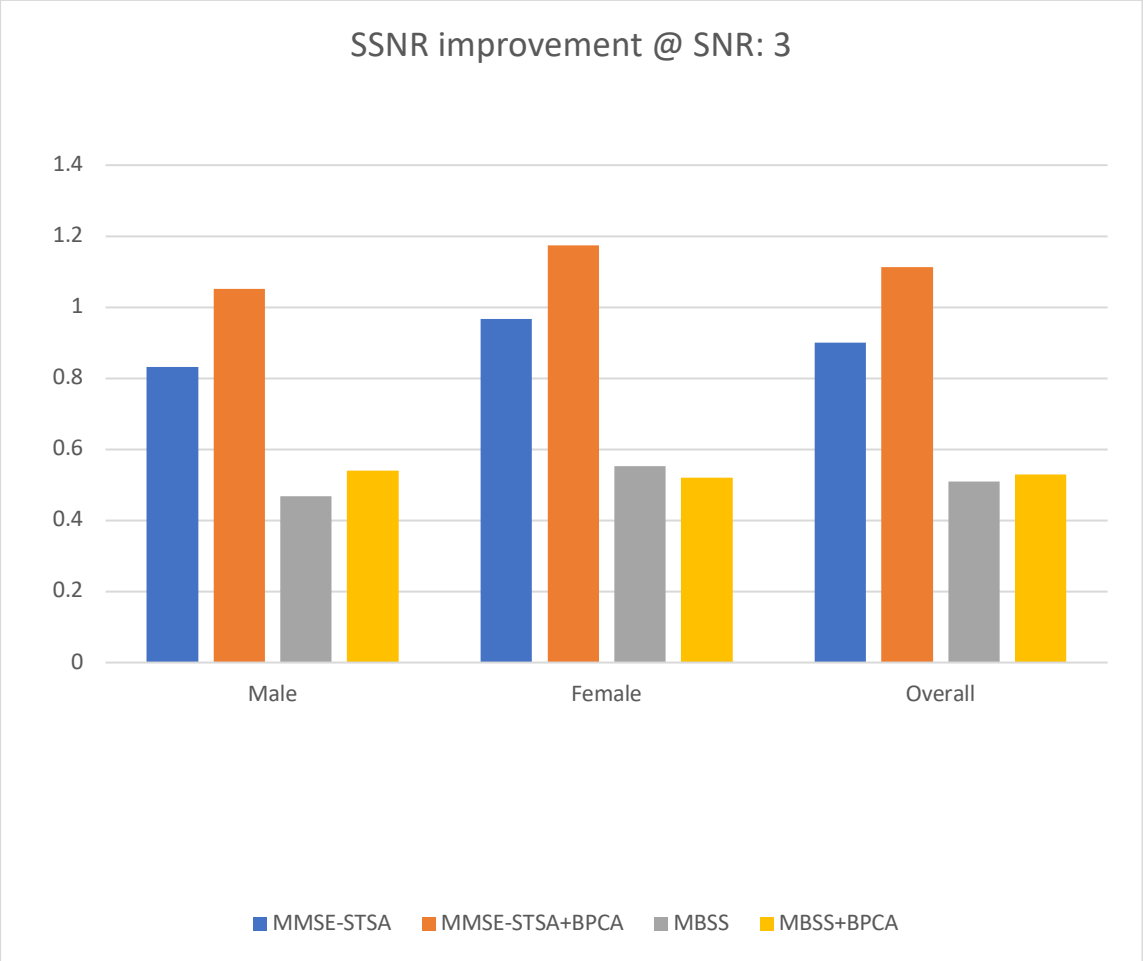


Figure 4.1.17: Average SSNR improvement in 3 dB with regard to the noise floor scores for male and female speakers under 'Babble' noise contamination [80].

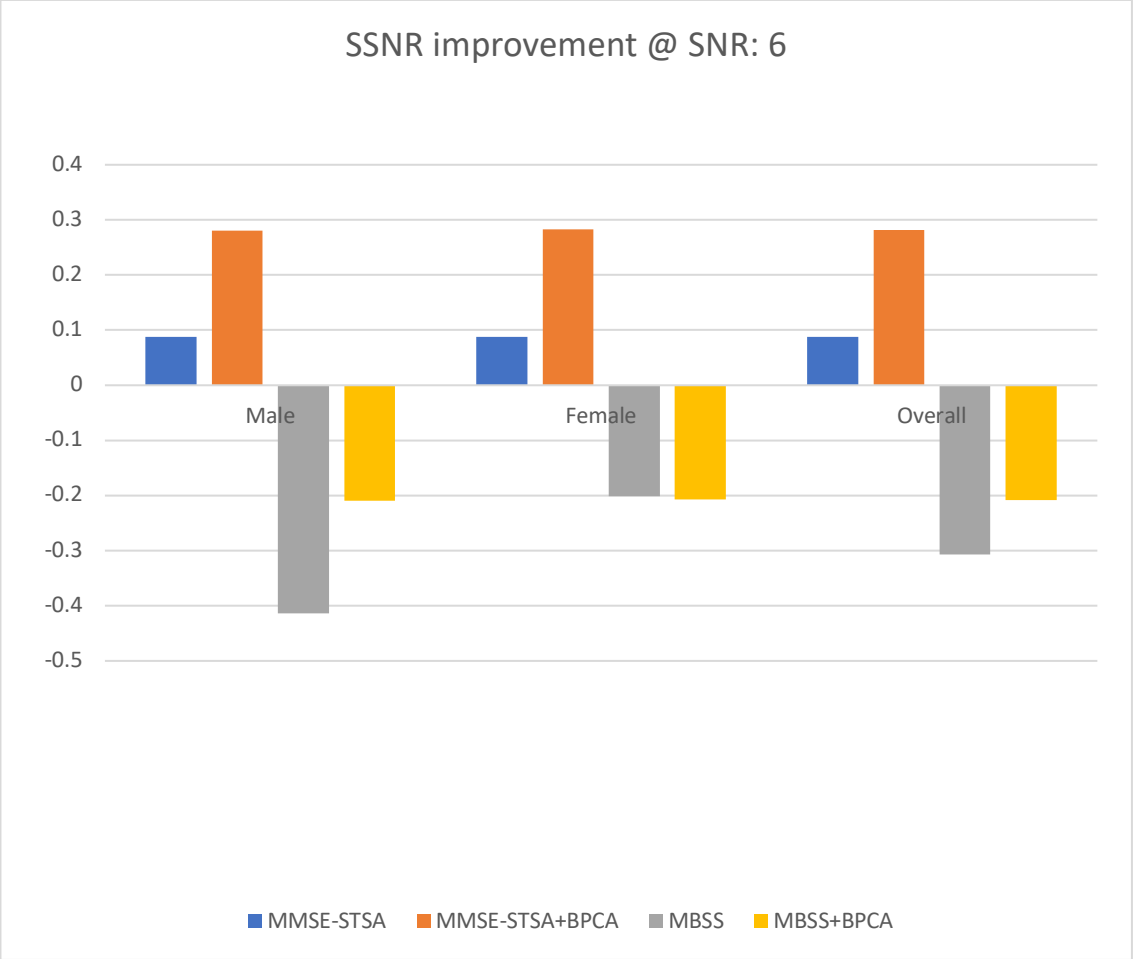


Figure 4.1.18: Average SSNR improvement in 6 dB with regard to the noise floor scores for male and female speakers under 'Babble' noise contamination [80].

4.2 Performance analysis under different noise types of contamination with -6 dB SNR

Due to the space constraint of the manuscript, in Table. 4.2, we have presented the average performance metrics (over 30 signals) for −6 dB SNR and noise types of Babble, Exhibition, and Car only. The results shown here further highlight efficacy of the proposed noise suppression algorithm under different noise types. Out of the three noise types shown here, scores for the "Car" noise type shows the highest gain in performance due to the inclusion Block-PCA w.r.t both MMSE-STSA and MBSS methods as well as the noisy

input signal. Moreover, for all noise types, the enhancement scores for the techniques with Block-PCA are higher than other methods emphasizing effectiveness of the proposed methodology.

4.3 Performance comparison for PESQ and SSNR over multiple noise types and SNR levels

In this section, we have presented the PESQ in Table. 4.3, and SSNR scores in Table. 4.4 for all 5 noise types and 6 SNR levels. The trend remains the same as seen in Tables. 4.1, and 4.2, i.e., the introduction of noise suppression in the enhancement framework helps the MMSE-STSA and MBSS methods in the recovery of improved speech signals. This performance gain is very significant when noise energy in the contaminated signal is high and gradually gets small as the noise levels get lower and lower. The results shown here are consistent with the MMSE-STSA + Block-PCA performing the best among other methods.

Table 4.2: Performance analysis for noise types Babble, Exhibition, and Car with -6 dB SNR level. Best results are highlighted in **BOLD** [80].

		Babble					
SNR dB	Method/Average Scores	PESQ	LLR	SSNR	Csig	Cbak	Covl
-6	Noisy	1.386	1.034	-7.152	2.11	1.301	1.616
	MMSE-STSA	1.302	1.158	-4.769	1.766	1.264	1.368
	MMSE-STSA + BPCA	**1.777**	1.095	**-4.237**	**2.115**	**1.513**	**1.754**
	MBSS	1.304	1.100	-5.208	1.898	1.285	1.440
	MBSS + BPCA	1.619	**1.049**	-5.110	2.107	1.413	1.675
		Exhibition					
SNR dB	Method Average Scores	PESQ	LLR	SSNR	Csig	Cloak	Covi
-6	Noisy	1.317	1.253	-7.198	1.911	1.3	1.496
	MMSE-STSA	1.261	1.245	-4.239	1.765	1.347	1.360
	MMSE-STSA + BPCA	**1.775**	**1.211**	**-3.891**	**2.091**	**1.606**	**1.767**
	MBSS	1.244	1.243	-4.779	1.790	1.33	1.373
	MBSS + BPCA	1.415	1.217	-4.606	1.930	1.437	1.525
		Car					
SNR dB	Method/Average Scores	PESQ	LLR	SSNR	Csig	Cloak	Covi
-6	Noisy	1.396	1.122	-7.434	2.057	1.3	1.581
	MMSE-STSA	1.599	1.083	-3.649	2.234	1.617	1.788
	MMSE-STSA + BPCA	**2.416**	**1.055**	**-3.506**	**2.736**	**2.011**	**2.436**
	MBSS	1.362	1.108	-4.656	2.029	1.561	1.561
	MBSS + BPCA	1.504	1.084	-4.542	2.151	1.685	1.685

Table 4.3: PESQ scores for all noise types and all SNR noise levels [80].

		PESQ					
SNR dB	Method/Noises	Airport	Babble	Car	Exhibition	Restaurant	Average
-9	Noisy	1.533	1.237	1.346	1.172	1.181	1.294
	MMSE-STSA	1.174	1.063	1.899	1.201	1.060	1.279
	MMSE-STSA + BPCA	1.910	1.860	2.732	1.986	1.456	**1.989**
	MBSS	1.256	1.178	1.236	1.128	1.122	1.184
	MBSS + BPCA	1.654	1.648	1.590	1.459	1.466	1.563
-6	Noisy	1.459	1.386	1.396	1.317	1.378	1.387
	MMSE-STSA	1.313	1.302	1.599	1.261	1.226	1.34
	MMSE-STSA + BPCA	1.746	1.777	2.416	1.775	1.552	**1.853**
	NBSS	1.403	1.304	1.362	1.244	1.324	1.327
	MBSS + BPCA	1.636	1.619	1.504	1.415	1.631	1.561
-3	Noisy	1.577	1.537	1.515	1.474	1.487	1.518
	MMSE-STSA	1.522	1.600	1.752	1.482	1.482	1.568
	MMSE-STSA + BPCA	1.748	1.800	1.982	1.729	1.692	**1.790**
	MBSS	1.618	1.587	1.598	1.424	1.586	1.563
	MBSS + BPCA	1.722	1.729	1.669	1.539	1.705	1.673
0	Noisy	1.753	1.742	1.662	1.614	1.767	1.708
	MMSE-STSA	1.822	1.858	1.904	1.718	1.696	1.799
	MMSE-STSA + BPCA	1.920	1.951	1.996	1.810	1.901	**1.916**
	MBSS	1.814	1.797	1.800	1.662	1.827	1.780
	MBSS + BPCA	1.905	1.890	1.897	1.774	1.889	1.871
3	Noisy	1.933	1.925	1.825	1.784	1.969	1.887
	MMSE-STSA	2.051	2.090	2.128	1.961	2.045	2.055
	MMSE-STSA + BPCA	2.104	2.144	2.178	2.017	2.113	**2.111**
	MBSS	2.047	2.015	2.005	1.894	2.013	1.995
	MBSS + BPCA	2.095	2.081	2.138	2.016	2.078	2.082
6	Noisy	2.114	2.105	2.000	1.965	2.145	2.066
	MMSE-STSA	2.275	2.277	2.338	2.190	2.242	2.264
	MMSE-STSA + BPCA	2.303	2.331	2.377	2.231	2.306	**2.310**
	MBSS	2.243	2.219	2.206	2.110	2.211	2.198
	MBSS + BPCA	2.280	2.272	2.346	2.228	2.265	2.278

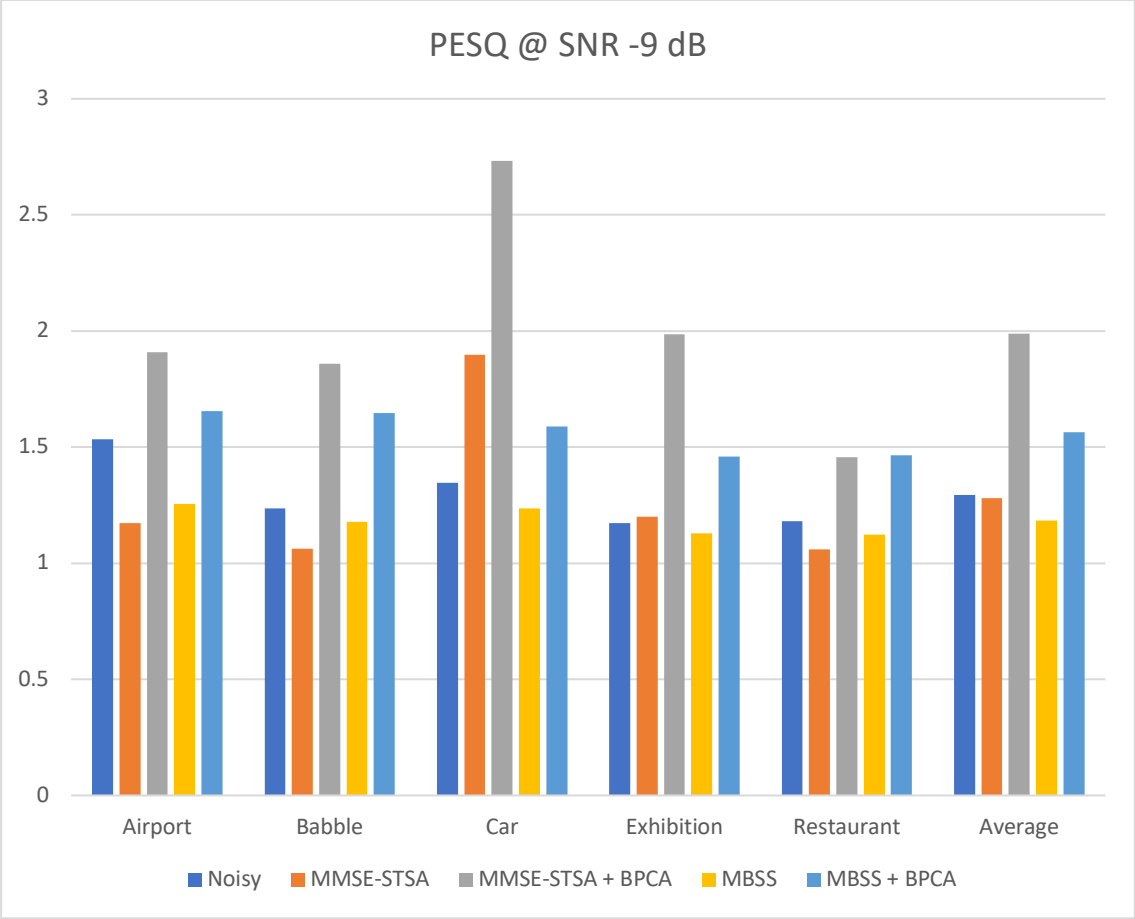


Figure 4.3.1: PESQ scores in -9 dB SNR for all noise types and all SNR noise levels [80].

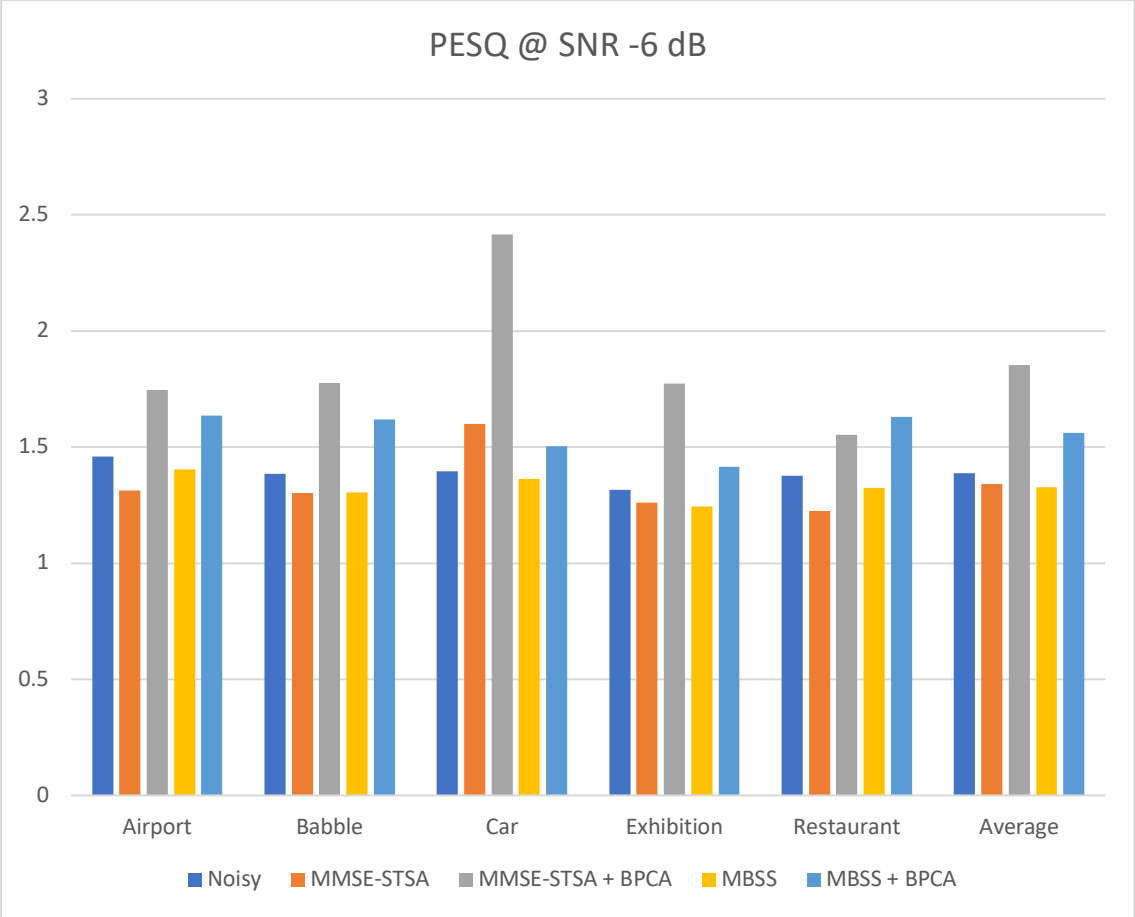


Figure 4.3.2: PESQ scores in -6 dB SNR for all noise types and all SNR noise levels [80].

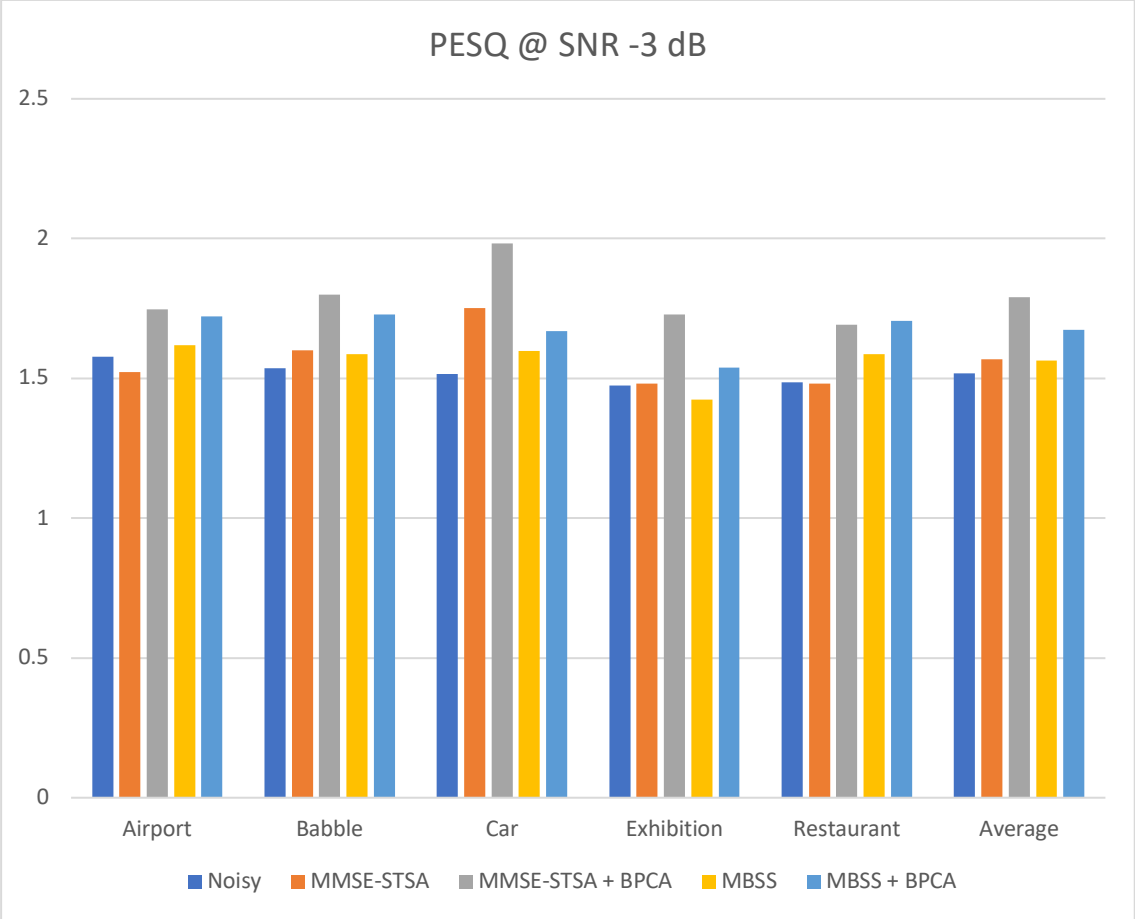


Figure 4.3.3: PESQ scores in -3 dB SNR for all noise types and all SNR noise levels [80].

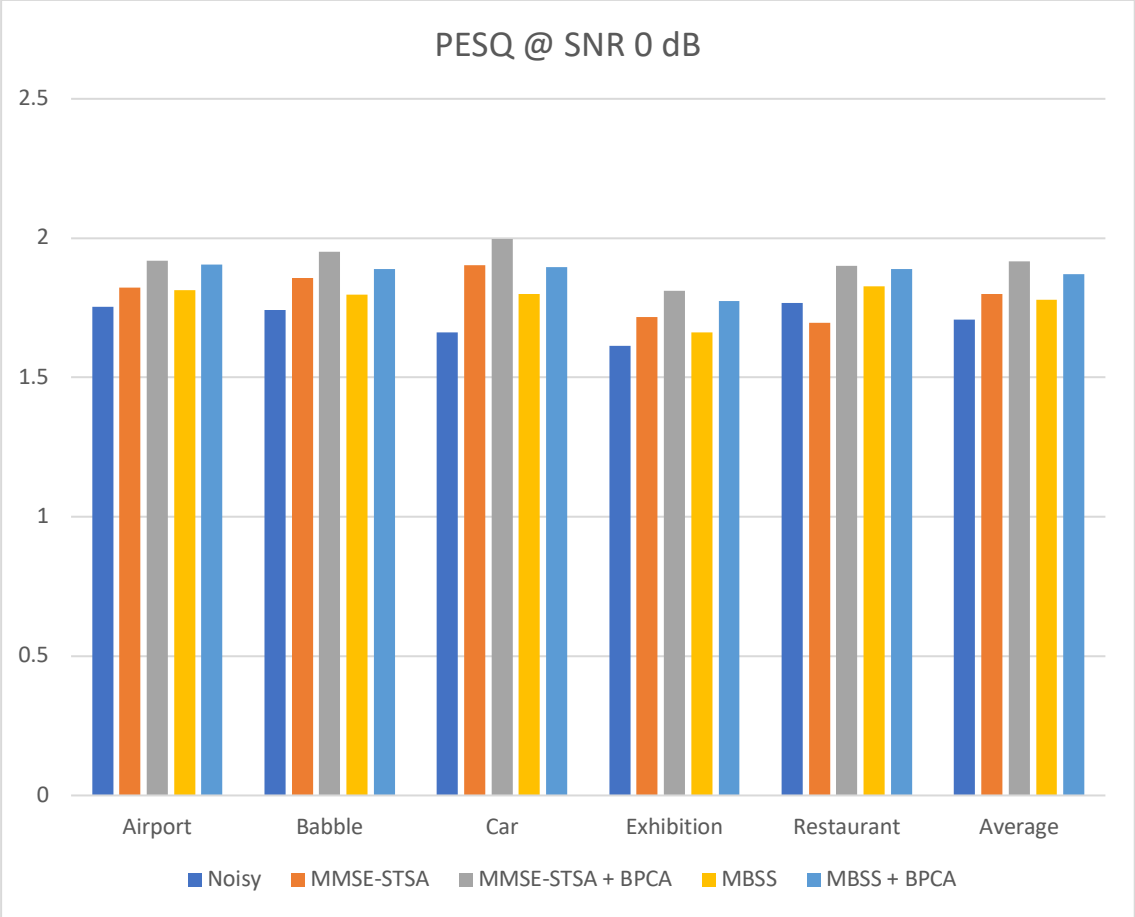


Figure 4.3.4: PESQ scores in 0 dB SNR for all noise types and all SNR noise levels [80].

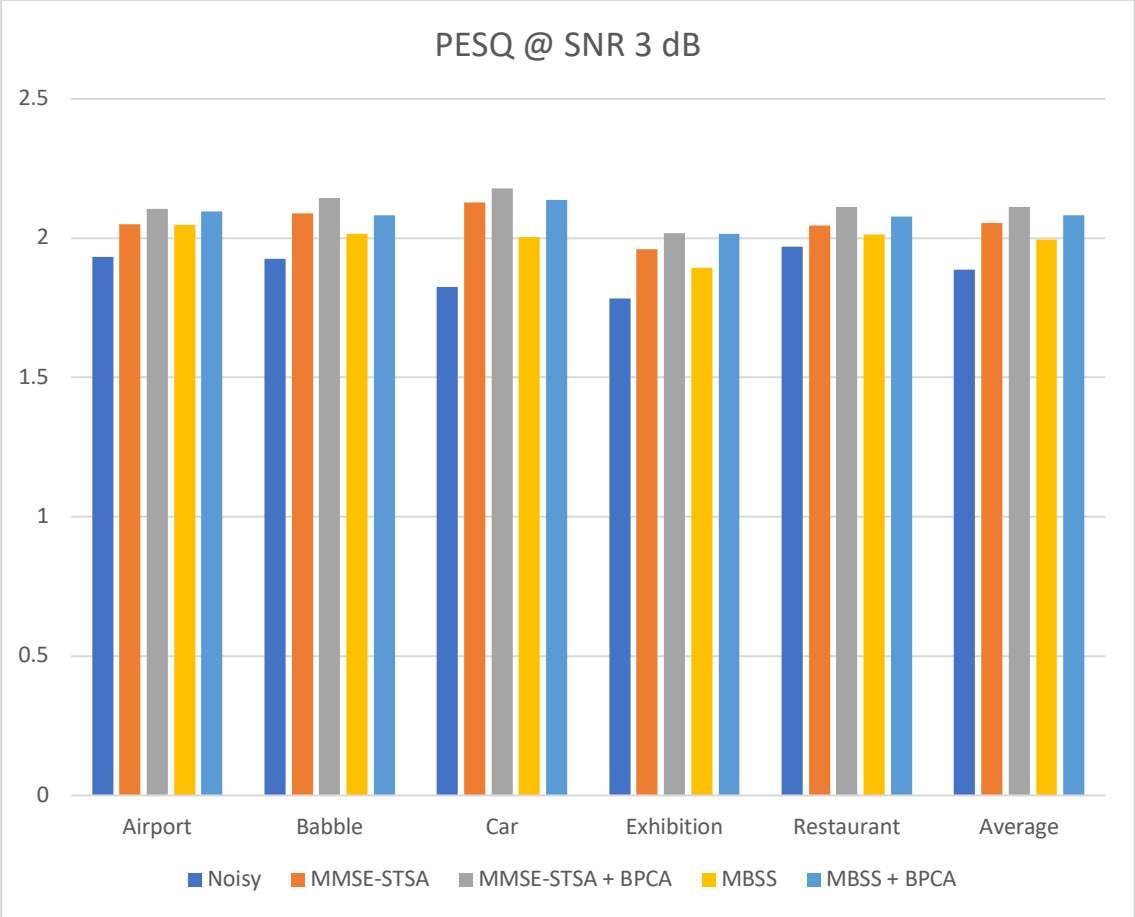


Figure 4.3.5: PESQ scores in 3 dB SNR for all noise types and all SNR noise levels [80].

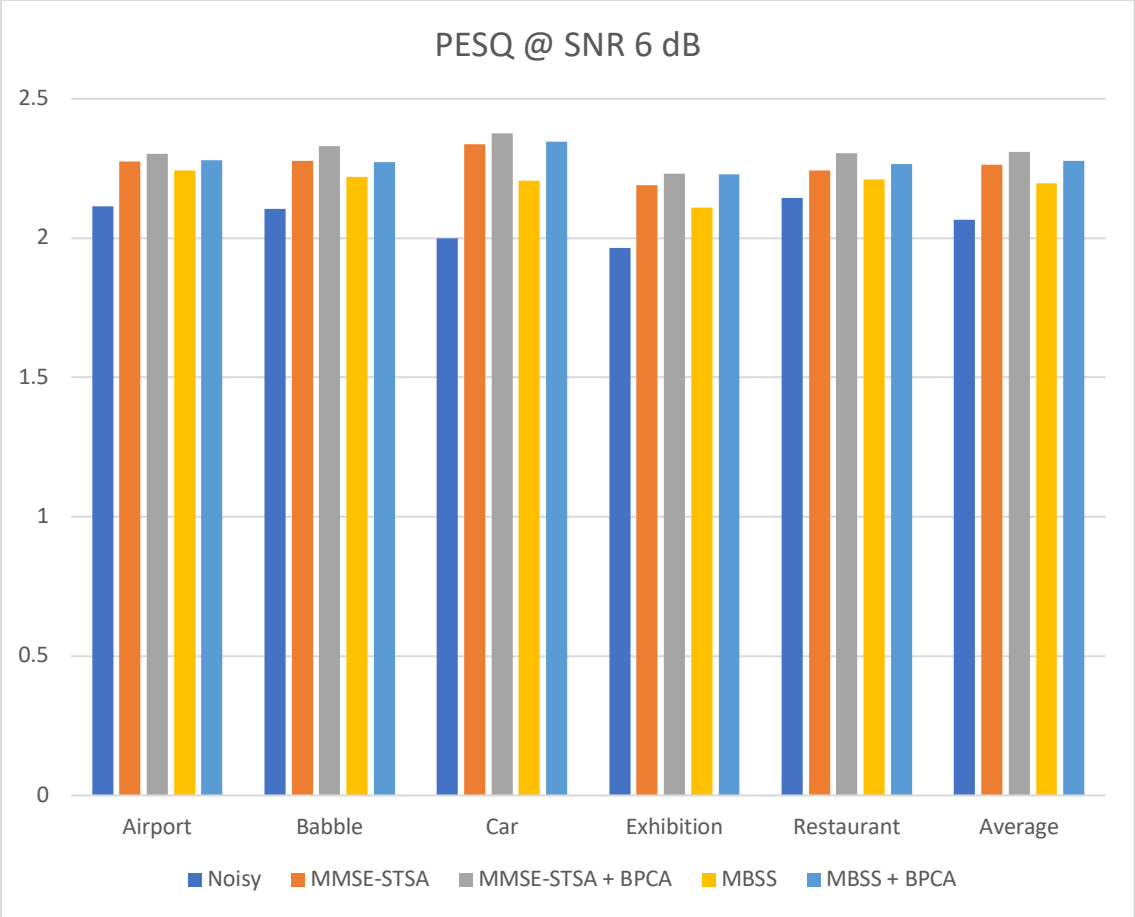


Figure 4.3.6: PESQ scores in 6 dB SNR for all noise types and all SNR noise levels [80].

Table 4.4: SSNR scores for all noise types and all SNR noise levels [80].

		SSNR					
SNR dB	Method/Noises	Airport	Babble	Car	Exhibition	Restaurant	Average
-9	Noisy	-8.091	-8.229	-8.475	-8.273	-7.910	-8.195
	MMSE-STSA	-5.310	-5.724	-4.395	-4.941	-5.774	-5.229
	MMSE-STSA + BPCA	-5.000	-5.063	-4.26	-4.611	-5.349	**-4.857**
	MBSS	-5.964	-6.113	-5.62	-5.731	-6.058	-5.897
	MBSS+BPCA	-5.907	-6.055	-5.532	-5.559	-6.055	-5.821
-6	Noisy	-6.984	-7.152	-7.434	-7.198	-6.790	-7.112
	MMSE-STSA	-4.488	-4.769	-3.649	-4.239	-4.930	-4.415
	MMSE-STSA + BPCA	-4.204	-4.237	-3.506	-3.891	-4.554	**-4.078**
	MBSS	-5.055	-5.208	-4.656	-4.779	-5.215	-4.982
	MBSS + BPCA	-5.045	-5.110	_4.542	-4.606	-5.247	-4.910
-3	Noisy	-5.656	-5.848	-6.165	-5.894	-5.441	-5.801
	MMSE-STSA	-3.611	-3.802	-2.901	-3.416	-3.985	-3.543
	MMSE-STSA + BPCA	-3.380	-3.454	-2.755	-3.144	-3.650	**-3.276**
	MBSS	-4.117	-4.188	-3.669	-3.843	-4.251	-4.013
	MBSS + BPCA	-4.110	-4.124	-3.546	-3.654	-4.310	-3.949
0	Noisy	-4.131	-4.352	-4.680	-4.388	-3.907	-4.292
	MMSE-STSA	-2.733	-2.805	-2.061	-2.540	-2.934	-2.615
	MMSE-STSA + BPCA	-2.452	-2.566	-1.908	-2.282	-2.689	**-2.379**
	MBSS	-3.143	-3.245	-2.701	-2.912	-3.278	-3.056
	MBSS + BPCA	-3.099	-3.205	-2.508	-2.697	-3.332	-2.968
3	Noisy	-2.449	-2.700	-3.029	-2.721	-2.221	-2.624
	MMSE-STSA	-1.681	-1.800	-1.190	-1.577	-1.882	-1.626
	MMSE-STSA + BPCA	-1.524	-1.587	-1.016	-1.326	-1.651	**-1.421**
	MBSS	-2.040	-2.190	-1.761	-1.977	-2.241	-2.042
	MBSS + BPCA	-2.060	-2.170	-1.453	-1.671	-2.255	-1.922
6	Noisy	-0.645	-0.912	-1.253	-0.925	-0.404	-0.828
	MMSE-STSA	-0.746	-0.824	-0.222	-0.607	-0.857	-0.651
	MMSE-STSA + BPCA	-0.611	-0.630	-0.041	-0.385	-0.662	**-0.466**
	MBSS	-1.002	-1.219	-0.794	-1.014	-1.212	-1.048
	MBSS + BPCA	-0.982	-1.120	-0.412	-0.658	-1.159	-0.866

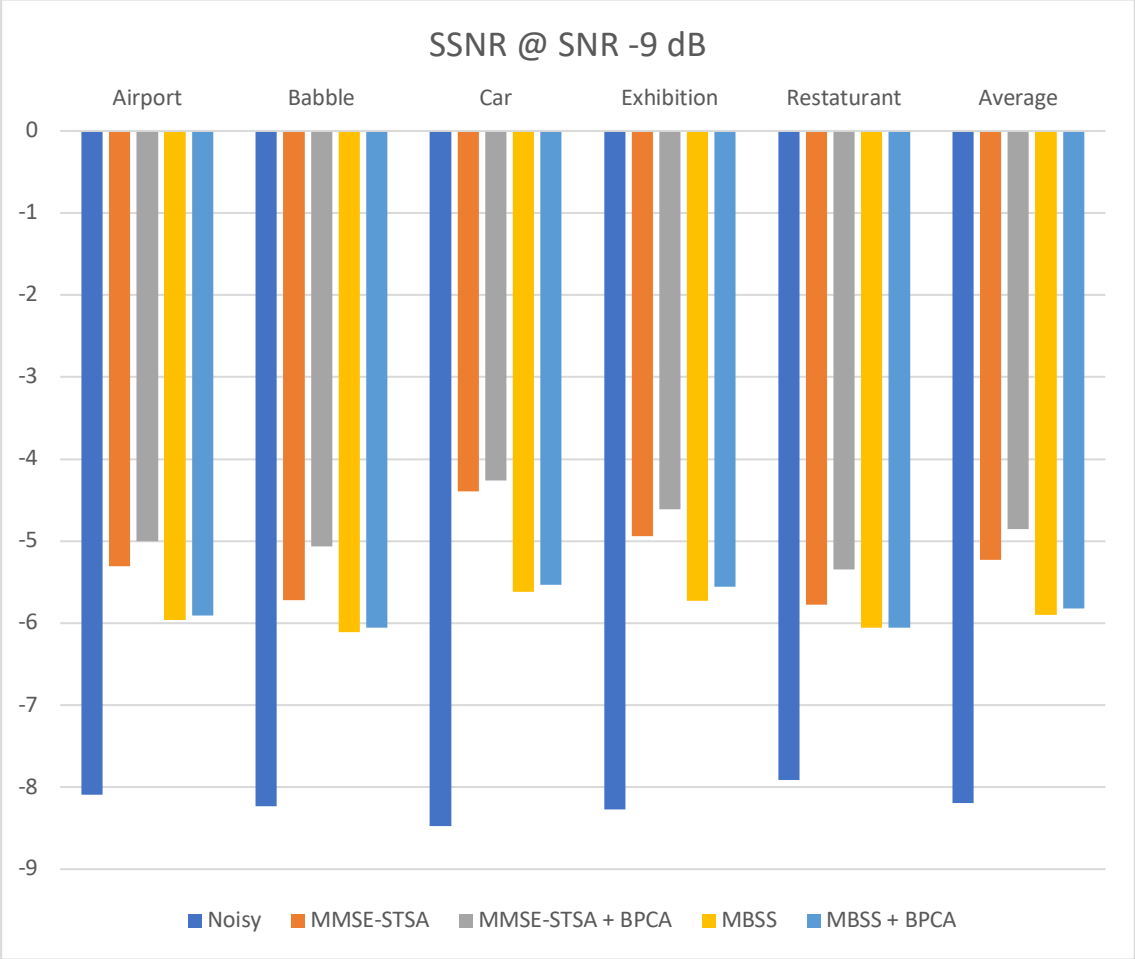


Figure 4.4.1: SSNR scores in -9 dB SNR for all noise types [80].

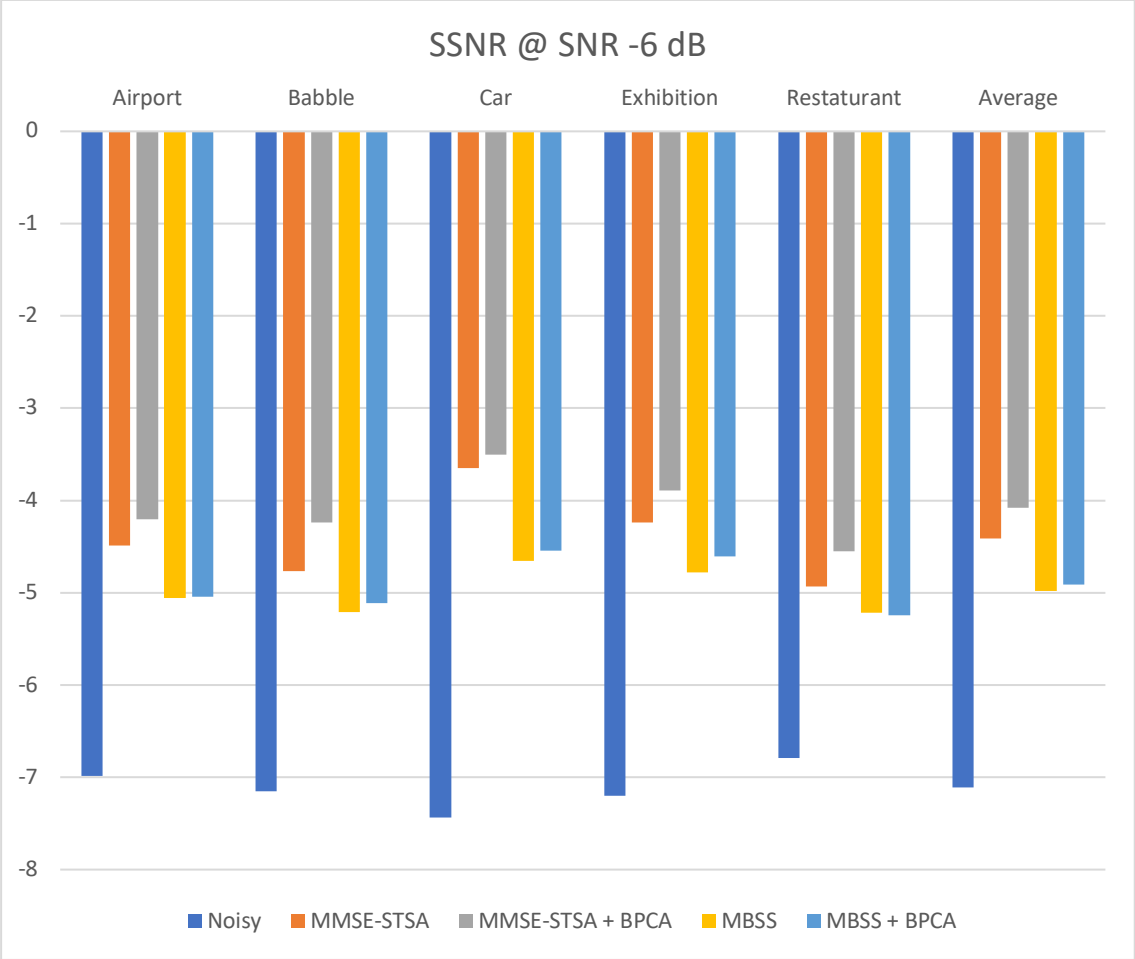


Figure 4.4.2: SSNR scores in -6 dB SNR for all noise types [80].

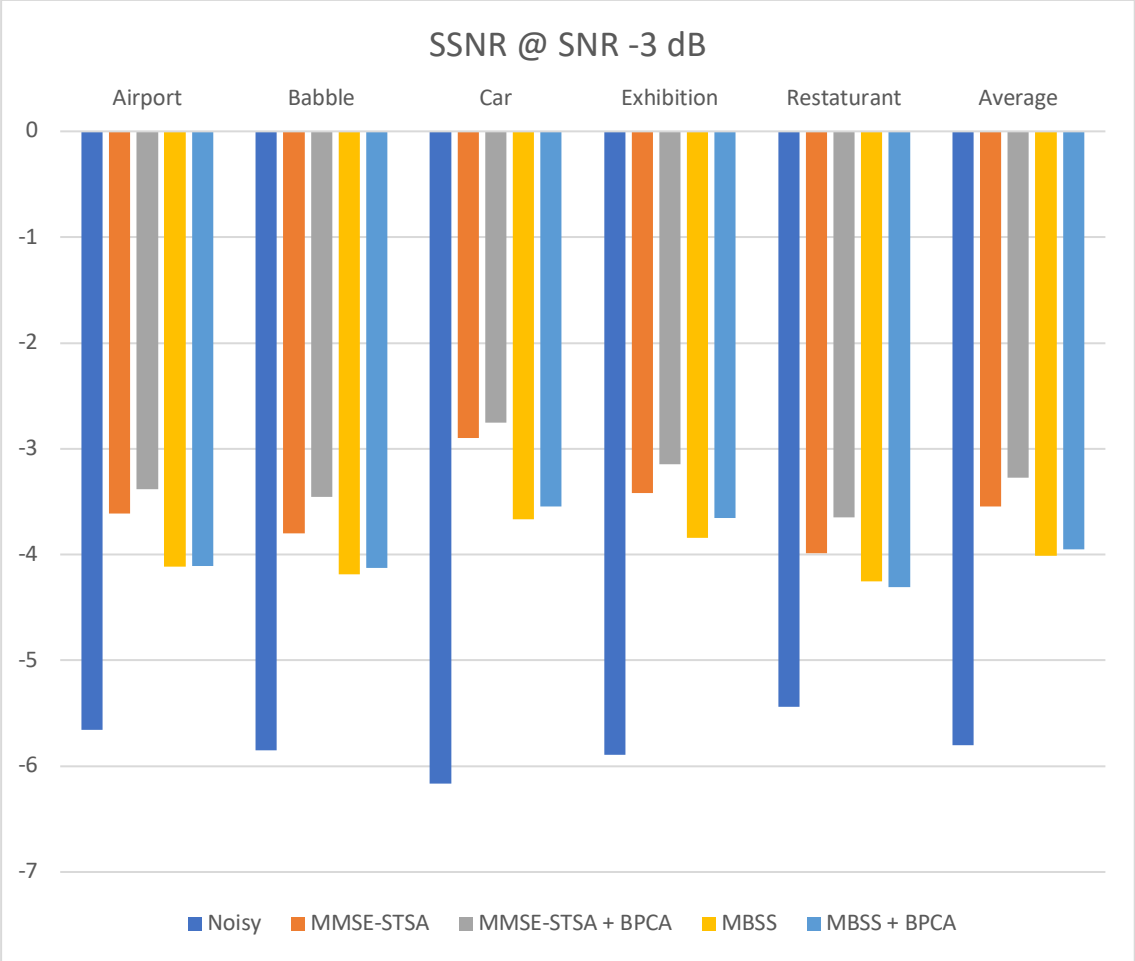


Figure 4.4.3: SSNR scores in -3 dB SNR for all noise types [80].

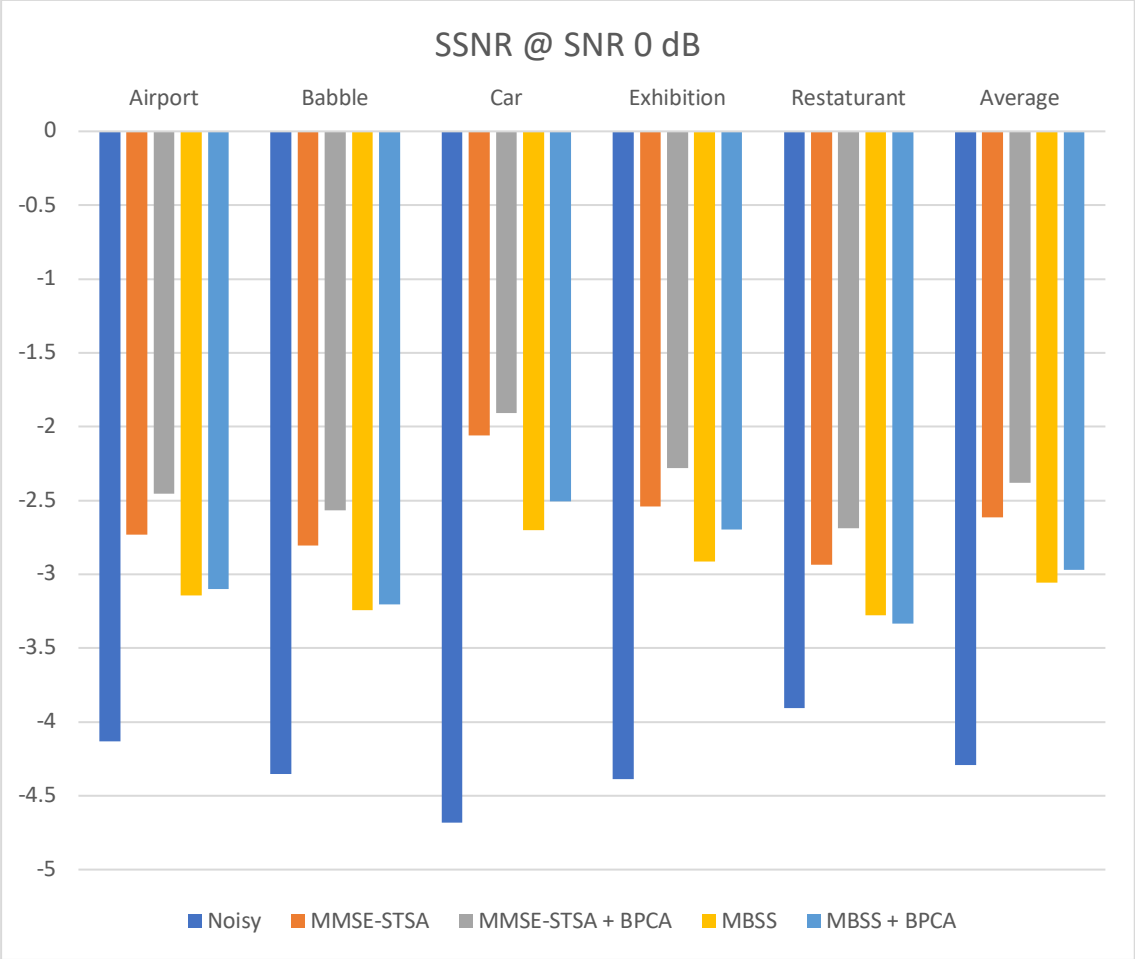


Figure 4.4.4: SSNR scores in 0 dB SNR for all noise types [80].

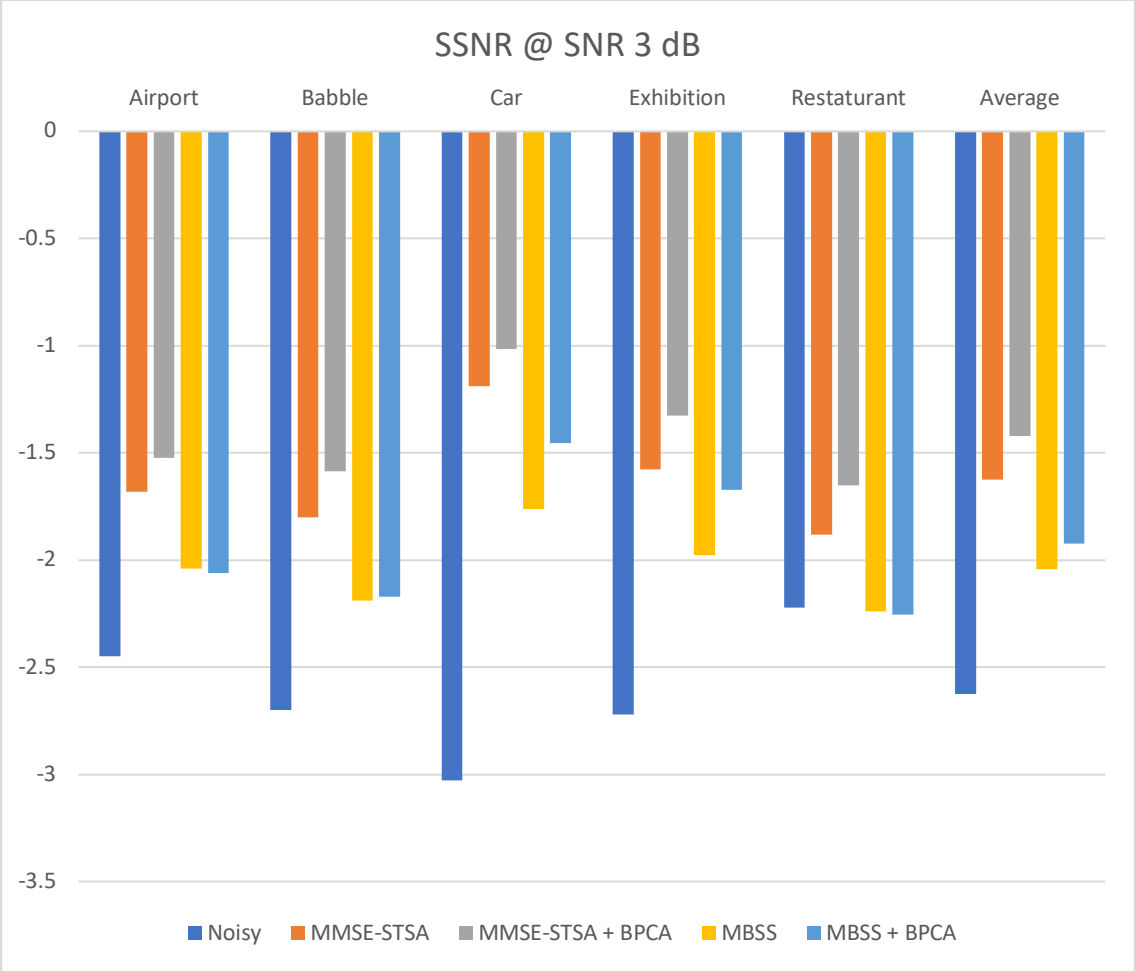


Figure 4.4.5: SSNR scores in 3 dB SNR for all noise types [80].

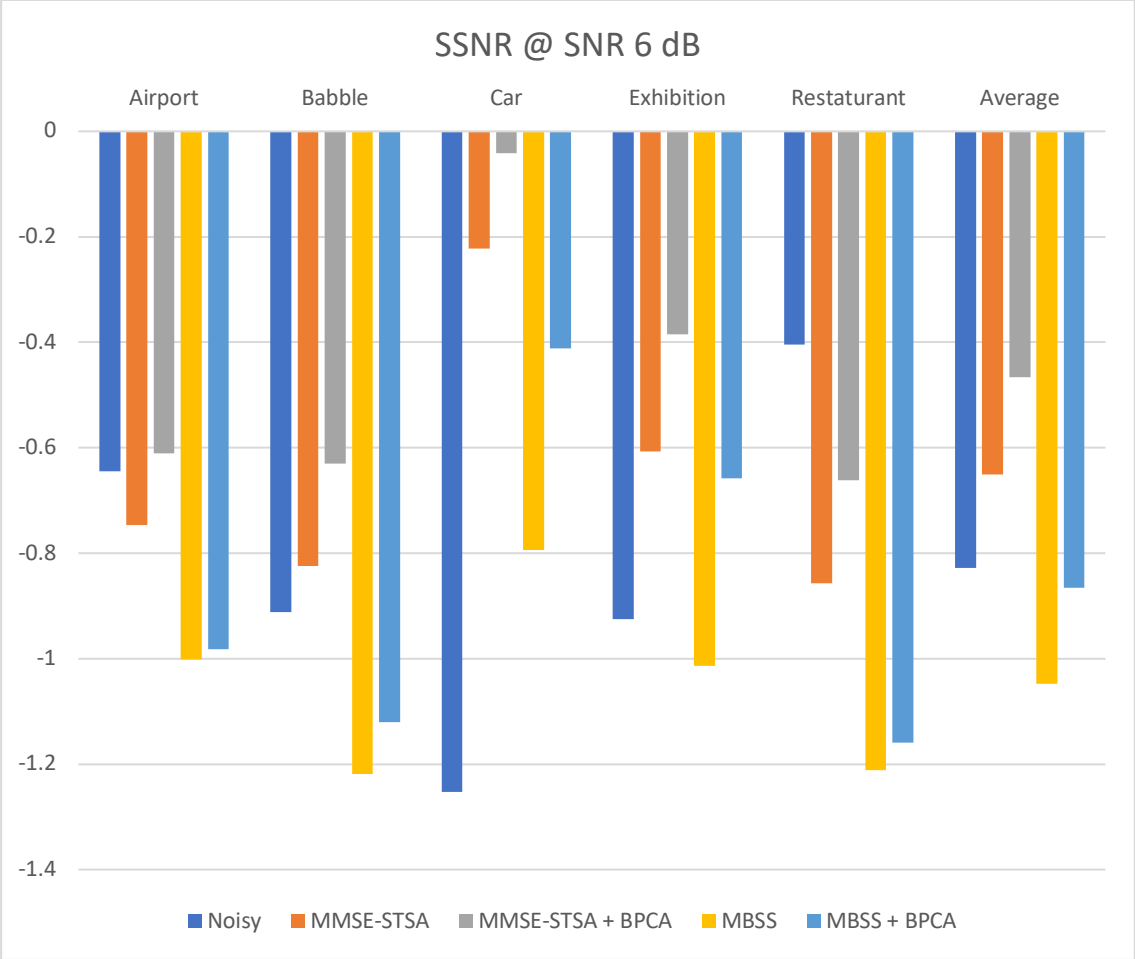


Figure 4.4.6: SSNR scores in 6 dB SNR for all noise types [80].

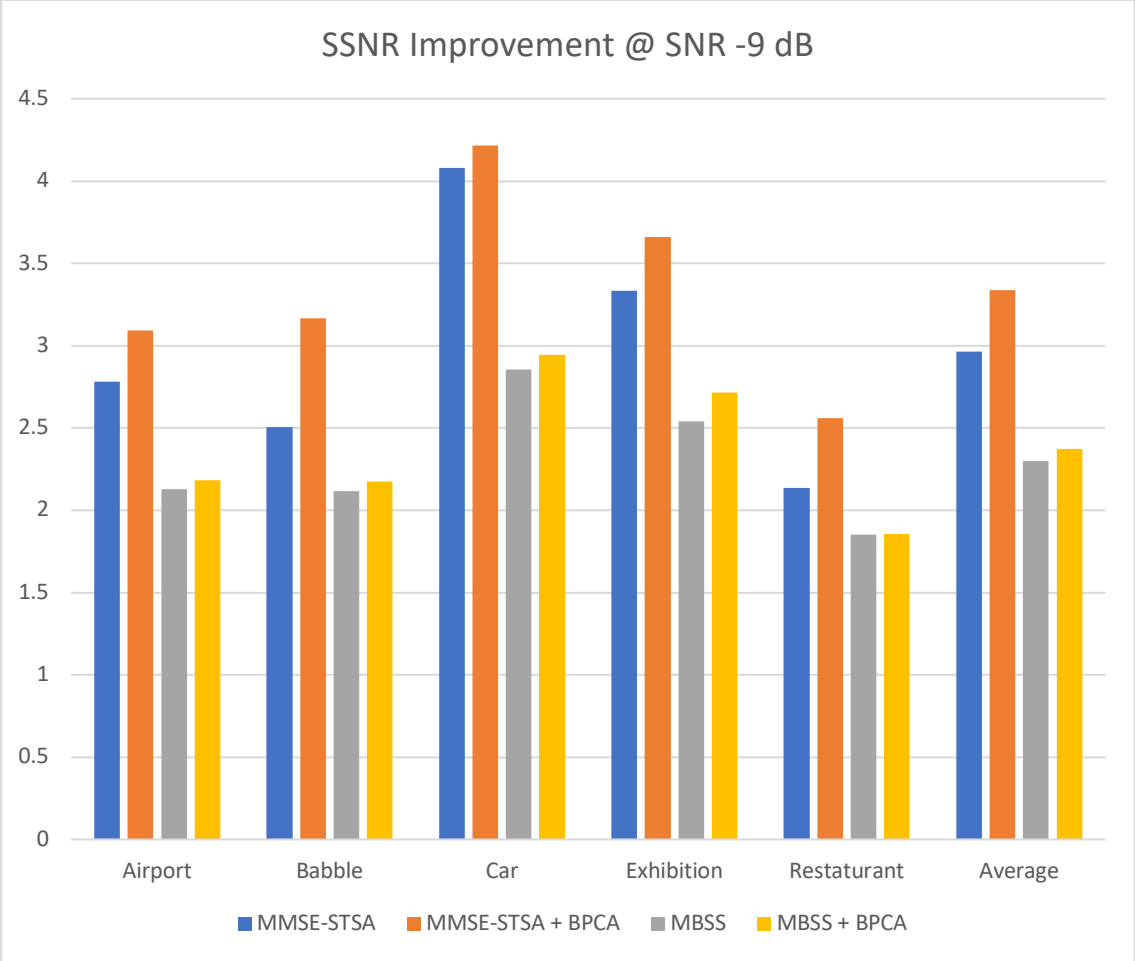


Figure 4.4.7: SSNR improvement in -9 dB with regard to the noise floor scores for all noise types [80].

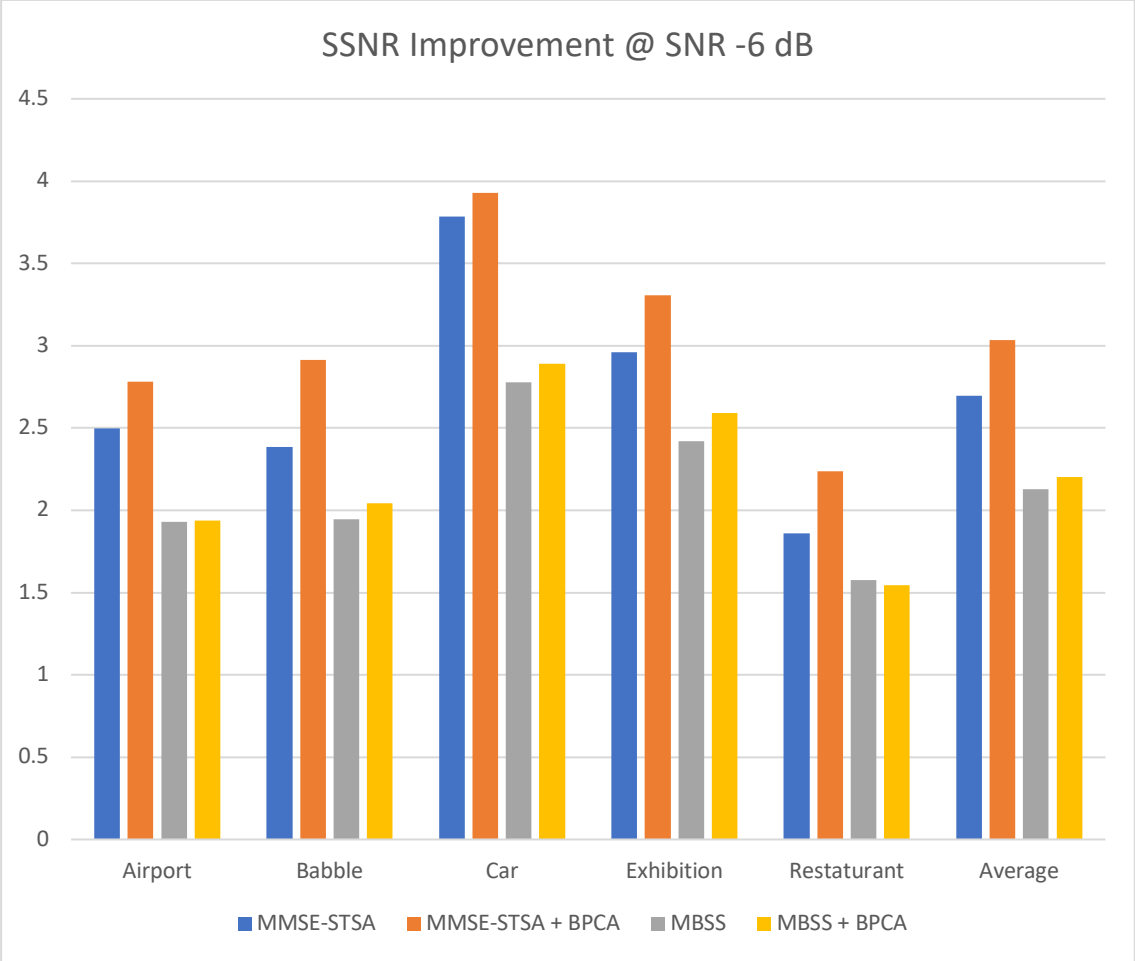


Figure 4.4.8: SSNR improvement in -6 dB SNR with regard to the noise floor scores for all noise types [80].

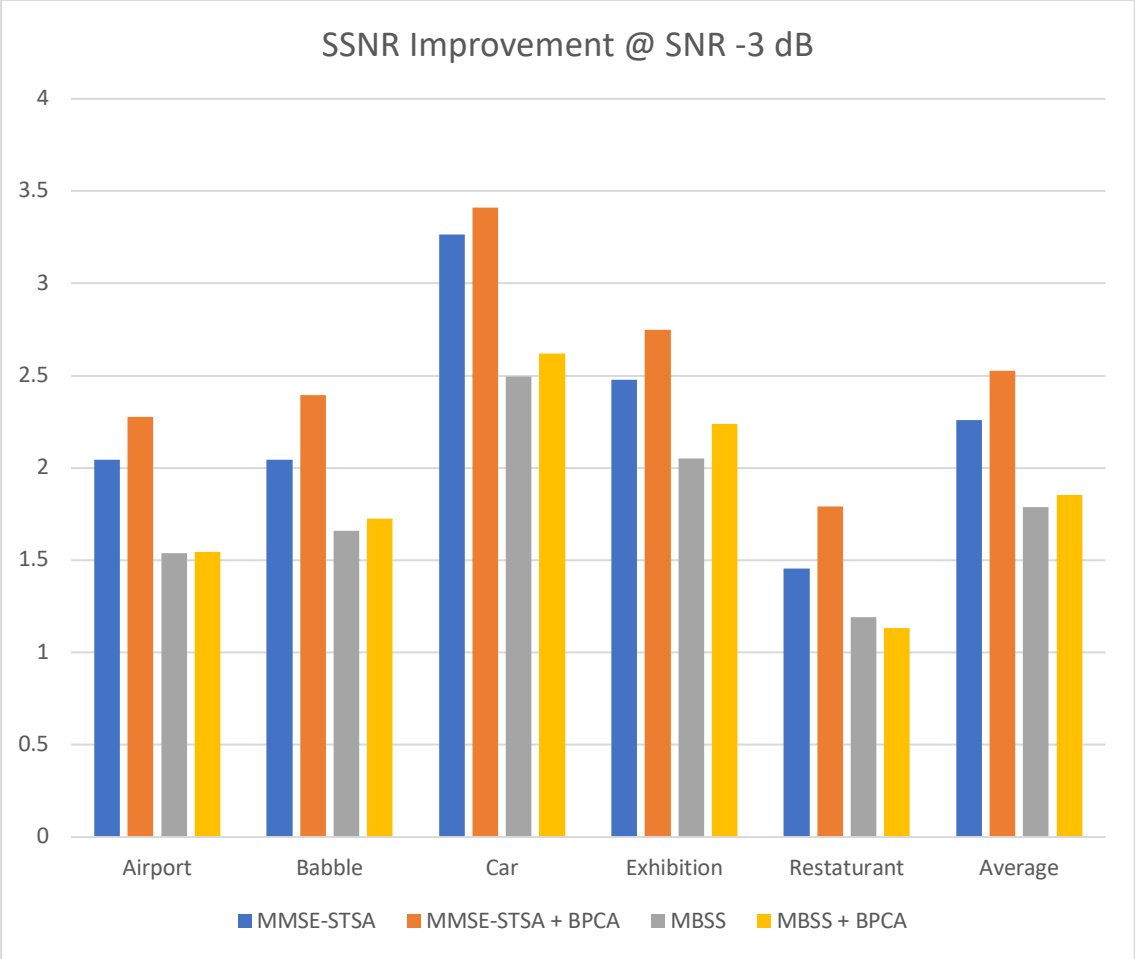


Figure 4.4.9: SSNR improvement in -3 dB SNR with regard to the noise floor scores for all noise types [80].

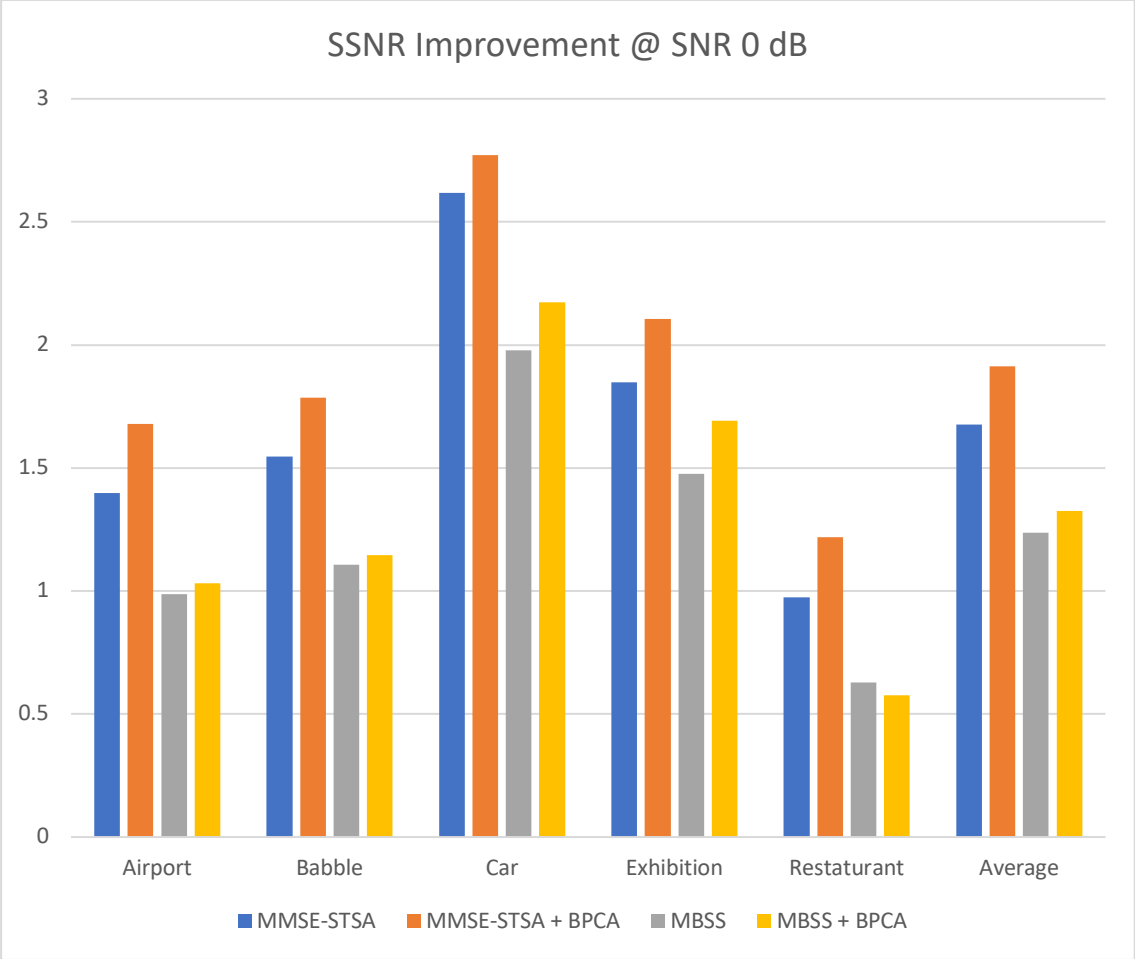


Figure 4.4.10: SSNR improvement in 0 dB SNR with regard to the noise floor scores for all noise types [80].

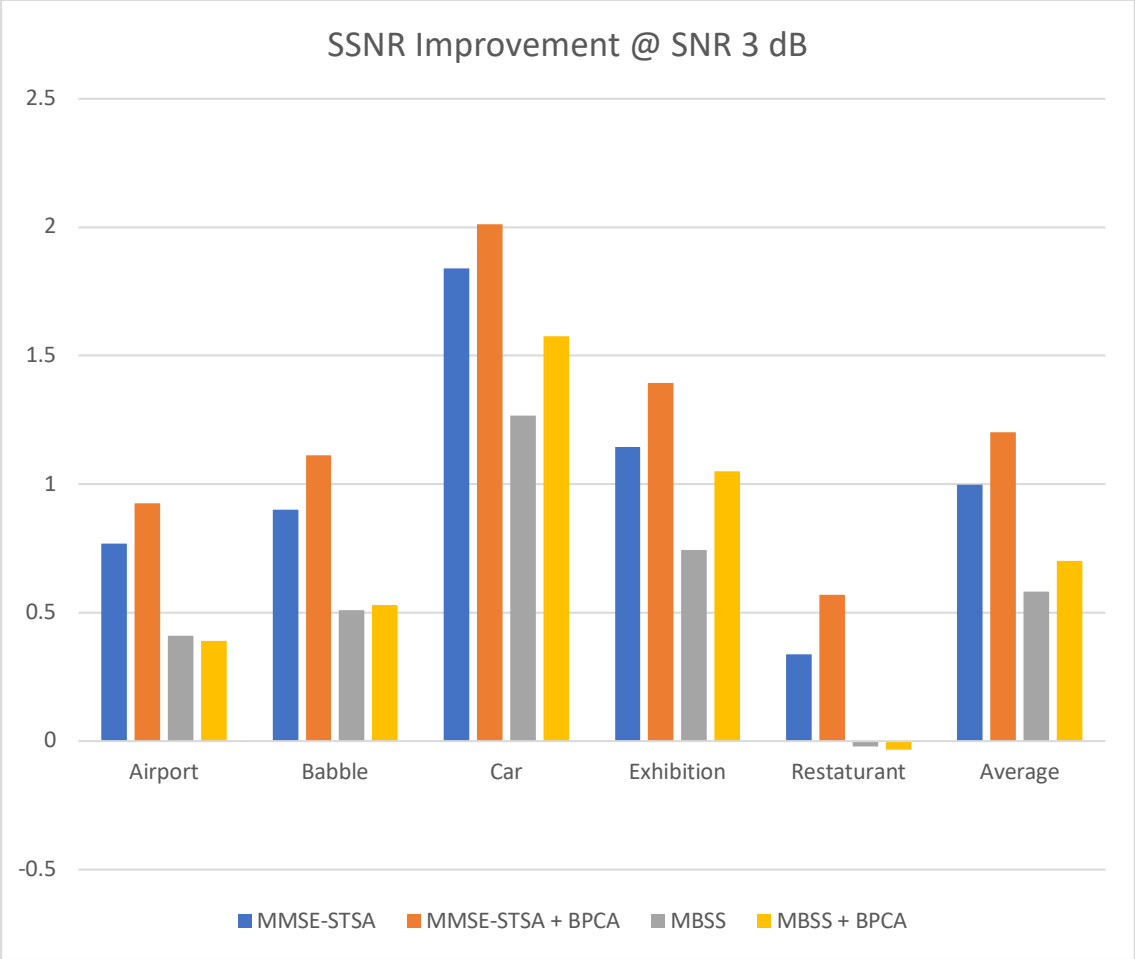


Figure 4.4.11: SSNR improvement in 3 dB SNR with regard to the noise floor scores for all noise types [80].

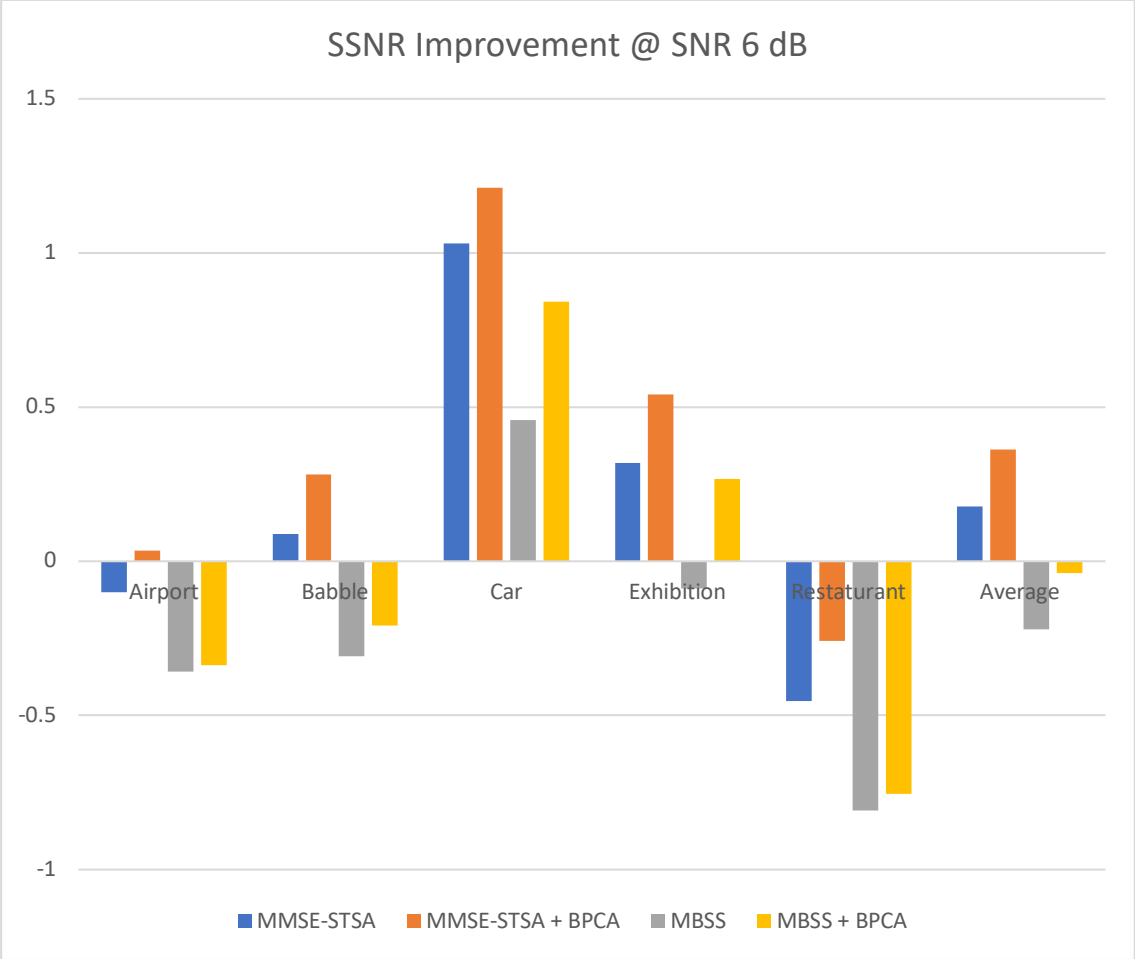


Figure 4.4.12:SSNR improvement in 6dB SNR with regard to the noise floor scores for all noise types [80].

4.4 Performance comparison for Babble and Exhibition noise types over multiple metrics and SNR levels

Finally in this section we summarize the score from all performance metrics for the noise types Babble and Exhibition in Tables 4.5 and 4.6 respectively. For both noise types, performance improvement resulting from inclusion of Block-PCA in the enhancement framework is evident across multiple metrics and is not sensitive to PESQ only. Moreover, the combination of MMSE-STSA + Block-PCA is performing better than the other comparative methods.

Table 4.5: All performance metric scores for noise type Babble over multiple SNR levels [80].

		Babble					
SNR dB	Method Noises	PESQ	LLR	SSNR	Csig	Cbak	Covl
-9	Noisy	1.237	1.106	-8.229	1.918	1.195	1.490
	MMSE-STSA	1.063	1.248	-5.724	1.459	1.097	1.136
	MMSE-STSA + BPCA	**1.860**	1.169	**-5.063**	**1.994**	**1.449**	**1.716**
	MBSS	1.178	1.172	-6.113	1.693	1.18	1.291
	MBSS + BPCA	1.648	**1.132**	-6.055	1.979	1.344	1.630
-6	Noisy	1.386	1.034	-7.152	2.110	1.301	1.616
	MMSE-STSA	1.302	1.158	-4.769	1.766	1.264	1.368
	MMSE-STSA + BPCA	**1.777**	1.095	**-4.237**	**2.115**	**1.513**	**1.754**
	MBSS	1.304	1.100	-5.208	1.898	1.285	1.440
	MBSS + BPCA	1.619	**1.049**	-5.110	2.107	1.413	1.675
-3	Noisy	1.537	0.950	-5.848	2.347	1.465	1.804
	MMSE-STSA	1.600	1.070	-3.802	2.125	1.512	1.687
	MMSE-STSA + BPCA	**1.800**	1.018	**-3.454**	2.311	**1.646**	**1.886**
	MBSS	1.587	1.010	-4.188	2.240	1.529	1.754
	MBSS + BPCA	1.729	**0.952**	-4.124	**2.356**	1.59	1.875
0	Noisy	1.742	0.857	-4.352	2.642	1.711	2.076
	MMSE-STSA	1.858	0.977	-2.805	2.462	1.765	2.009
	MMSE-STSA + BPCA	**1.951**	0.934	**-2.566**	2.571	**1.837**	2.113
	MBSS	1.797	0.916	-3.245	2.533	1.743	2.026
	MBSS + BPCA	1.890	**0.859**	-3.205	**2.631**	1.790	**2.120**
3	Noisy	1.925	0.757	-2.700	2.933	1.963	2.334
	MMSE-STSA	2.090	0.881	-1.800	2.788	2.007	2.313
	MMSE-STSA + BPCA	**2.144**	**0.835**	**-1.587**	2.881	**2.059**	2.390
	MBSS	2.015	0.812	-2.190	2.849	1.974	2.315
	MBSS + BPCA	2.081	0.757	-2.170	**2.940**	2.005	**2.392**
6	Noisy	2.105	0.654	-0.912	3.223	2.220	2.591
	MMSE-STSA	2.277	0.776	-0.824	3.091	2.222	2.582
	MMSE-STSA + BPCA	**2.331**	**0.742**	**-0.630**	3.168	**2.272**	**2.648**
	MBSS	2.219	0.711	-1.219	3.142	2.184	2.582
	MBSS + BPCA	2.272	0.664	-1.120	**3.226**	2.222	2.652

Table 4.6: All performance metric scores for noise type Exhibition over multiple SNR levels [80].

		Exhibition					
SNR dB	Method/Noises	PESQ	LLR	SSNR	Csig	Cbak	Covl
-9	Noisy	1.172	1.319	-8.273	1.705	1.18	1.347
	MMSE-STSA	1.201	1.326	-4.941	1.595	1.26	1.296
	MMSE-STSA + BPCA	**1.986**	1.304	**-4.611**	**2.048**	**1.606**	**1.843**
	MBSS	1.128	1.313	-5.731	1.597	1.22	1.273
	MBSS + BPCA	1.459	**1.298**	-5.559	1.815	1.37	1.500
-6	Noisy	1.317	1.253	-7.198	1.911	1.3	1.496
	MMSE-STSA	1.261	1.245	-4.239	1.765	1.347	1.360
	MMSE-STSA + BPCA	**1.775**	**1.211**	**-3.891**	**2.091**	**1.606**	**1.767**
	MBSS	1.244	1.243	-4.779	1.79	1.33	1.373
	MBSS + BPCA	1.415	1.217	-4.606	1.93	1.437	1.525
-3	Noisy	1.474	1.169	-5.894	2.154	1.481	1.696
	MMSE-STSA	1.482	1.141	-3.416	2.077	1.555	1.633
	MMSE-STSA + BPCA	**1.729**	1.111	**-3.144**	**2.243**	**1.688**	**1.836**
	MBSS	1.424	1.15	-3.843	2.058	1.521	1.600
	MBSS + BPCA	1.539	**1.108**	-3.654	2.177	1.598	1.720
0	Noisy	1.614	1.069	-4.388	2.406	1.693	1.910
	MMSE-STSA	1.718	1.044	-2.54	2.377	1.768	1.915
	MMSE-STSA + BPCA	**1.810**	1.011	**-2.282**	2.454	**1.829**	1.998
	MBSS	1.662	1.040	-2.912	2.378	1.742	1.896
	MBSS + BPCA	1.774	**0.979**	-2.697	**2.517**	1.82	**2.023**
3	Noisy	1.784	0.958	-2.721	2.687	1.93	2.154
	MMSE-STSA	1.961	0.945	-1.577	2.687	1.993	2.210
	MMSE-STSA + BPCA	**2.017**	0.914	**-1.326**	2.749	2.035	2.267
	MBSS	1.894	0.923	-1.977	2.698	1.958	2.189
	MBSS + BPCA	2.016	**0.852**	-1.671	**2.855**	**2.050**	**2.330**
6	Noisy	1.965	0.841	-0.925	2.977	2.177	2.407
	MMSE-STSA	2.19	0.842	-0.607	2.996	2.214	2.497
	MMSE-STSA + BPCA	**2.231**	0.818	**-0.385**	3.040	2.246	2.537
	MBSS	2.11	0.802	-1.014	3.012	2.168	2.471
	MBSS + BPCA	2.228	**0.736**	-0.658	**3.166**	**2.263**	**2.611**

Chapter Five: Conclusions and Discussion

The aim of this dissertation is to develop an unsupervised learning-based speech enhancement methodology for applications that require computationally lightweight and low-power solutions, such as assisted hearing aids, cochlear implants, and voice commands. The aim is to not only to improve the speech quality of the noisy signal, but also to improve its intelligibility. The objective is to achieve performance improvement without incorporating high computational complexity, increase in power requirements, and without performing offline learning through databases. Considering these constraints, this dissertation has investigated how much performance improvement can be achieved using only unsupervised statistical speech denoising algorithms. only. Several computationally and power efficient algorithms are already available in the literature. Existing techniques have achieved better speech enhancement performance for cases of correlated as well as statistically independent additive noise contamination. But a major challenge and key issue with all these algorithms is their susceptibility to the input SNR levels, i.e., under moderate to high SNR, their enhancement results are objectively better, however, under low to poor SNR conditions, these methods fail, altogether.

Therefore, the focus is on proposing a method to tackle and overcome challenges when low to poor input SNR scenarios are considered by developing a ‘technique’ that improves the input SNR. This is achieved by incorporating a pre-processing step into the entire speech enhancement system. This pre-processing step involves a variation of PCA,

called Block-PCA, which operates on blocks of short-time Fourier transform (STFT) of the noisy signal and generates an STFT approximation where noise levels have been suppressed. This approximation is then input to other speech enhancement algorithms to extract the enhanced speech signal. The specific approaches implemented in all experiments use the minimum mean square error - short-time spectral amplitude (MMSE-STSA) estimator [8] and the multi band spectral subtraction (MBSS) method [21].

For experimental evaluation, we used the NOIZEUS [78] database, which contains many voice samples with multiple speakers. The 5 noise types available in NOIZEUS database are considered and combined with clean speech signal to generate 6 different SNR levels for evaluation. In all cases, performance metrics, and SNR levels considered in Chapter 4, have shown that inclusion of the proposed pre-processing step leads to performance improvement in both MMSE-STSA and MBSS algorithms across all noise types, noise levels, and evaluation metrics. Comparisons are against approaches when the pre-processing is not included/considered. Thus, it is stated with confidence that the introduced pre-processing step/algorithm has been crucial to achieving performance enhancement.

5.1 Contributions and clarifications

The unique aspects and contributions of this research are summarized below:

i. The proposed unsupervised learning algorithm is computationally less expensive. It consumes less power, and it is easier to implement using only one microphone.

ii. The proposed method is suitable for devices that require a long battery life with less electronic components, such as hearing aid devices.

iii. The chosen PCA method is chosen over a band pass filter because of considering the whole signal from 0 to 4 kHz, containing both the voice and the noise signals. The proposed approach eliminates the additive noise signal in the whole frequency range, as opposed to the band pass filter that centers on a specific frequency range ($f_L - f_H$). Consequently, the followed approach has an advantage over band pass filters as it is capable of effectively removing noise signals in the voice band, which band pass filters are unable to eliminate.

iv. Rather than computing a single SNR for the whole signal, the input signal is segmented, and then, several SNR values are calculated. This is the Segmented SNR (SSNR). The SSNR has an advantage that it is easier to compare, evaluate, and to interpret the quality of audio signals.

v. PESQ is used to evaluate the quality of voice signals. It is a standardized method for perceiving the quality of such speech signals. It provides a single number rating from -0.5 to 4.5 where the higher score indicates better voice quality.

5.2 Discussion and Future Work

The model, framework, and methods considered in this dissertation focus on single microphone or channel based speech enhancement systems. They are statistical methods. However, obtained results may be generalized. For example, based on the discussion in Chapter 1, the proposed framework may be extended for multi-microphone/channel speech enhancement systems by adopting the same methods at the output or input of the beamformer. The performance of the pre-processing step may be further be improved by using sparse representation based methods [49, 50, 51, 52], instead of PCA for the STFT

approximation. In addition, the proposed pre-processing based formulation may be generalized and may be implemented using supervised learning techniques, as well, under severe noise degradation cases.

References

[1] P. C. Loizou, Speech enhancement: theory and practice. CRC press, 2013.

[2] Hendriks, R. Heusdens, U. Kjems, and J. Jensen, “On optimal multichannel meansquared error estimators for speech enhancement,” IEEE Signal Processing Letters, vol. 16, no. 10, pp. 885–888, 2009.

[3] Le Bouquin-JeannÃ¨s, P. Scalart, G. Faucon, and C. Beaugeant, “Combined noise and echo reduction in hands-free systems: A survey,” IEEE Trans., Speech, Audio, Process, vol. 9, no. 8, pp. 808–820, 2001.

[4] Hendriks, T. Gerkmann, and J. Jensen, “Dft-domain based single-microphone noise reduction for speech enhancement: A survey of the state of the art,” Synthesis Lectures on Speech and Audio Processing, vol. 9, no. 1, pp. 1–80, 2013.

[5] R. McAulay and M. Malpass, “Speech enhancement using a soft-decision noise suppression filter,” IEEE transactions on acoustics, speech, and signal processing, vol. 28, no. 2, pp. 137–145, 1980.

[6] T. Lotter and P. Vary, “Speech enhancement by map spectral amplitude estimation using a super-gaussian speech model,” EURASIP Journal on Advances in Signal Processing, vol. 2005, no. 7, pp. 1–17, 2005.

[7] P. J. Wolfe and S. J. Godsill, “Simple alternatives to the ephraim and malah suppression rule for speech enhancement,” in Proceedings of the 11th IEEE Signal Processing Workshop on Statistical Signal Processing (Cat. No. 01TH8563), pp. 496–499, IEEE, 2001.

[8] Y. Ephraim and D. Malah, “Speech enhancement using a minimum-mean square error short-time spectral amplitude estimator,” IEEE Transactions on acoustics, speech, and signal processing, vol. 32, no. 6, pp. 1109–1121, 1984.

[9] Y. Ephraim and D. Malah, “Speech enhancement using a minimum mean-square error log-spectral amplitude estimator,” IEEE transactions on acoustics, speech, and signal processing, vol. 33, no. 2, pp. 443–445, 1985.

[10] C. Breithaupt and R. Martin, “Mmse estimation of magnitude-squared dft coefficients with supergaussian priors,” in 2003 IEEE International Conference on Acoustics, Speech, and Signal Processing, 2003. Proceedings.(ICASSP’03)., vol. 1, pp. I–I, IEEE, 2003.

[11] I. Cohen, “Speech enhancement using super-gaussian speech models and noncausal a priori snr estimation,” Speech communication, vol. 47, no. 3, pp. 336– 350, 2005.

[12] B. Chen and P. C. Loizou, “A laplacian-based mmse estimator for speech enhancement,” Speech communication, vol. 49, no. 2, pp. 134–143, 2007.

[13] J. S. Erkelens, R. C. Hendriks, R. Heusdens, and J. Jensen, “Minimum meansquare error estimation of discrete fourier coefficients with generalized gamma priors,” IEEE Transactions on Audio, Speech, and Language Processing, vol. 15, no. 6, pp. 1741–1752, 2007.

[14] P. C. Loizou, “Speech enhancement based on perceptually motivated Bayesian estimators of the magnitude spectrum,” IEEE Transactions on Audio, Speech, and Language Processing, vol. 13, no. 5, pp. 857–869, 2005.

[15] R. Martin, “Speech enhancement based on minimum mean-square error estimation and supergaussian priors,” IEEE Transactions on Audio, Speech, and Language Processing, vol. 13, no. 5, pp. 845–856, 2005.

[16] I. Cohen, “Optimal speech enhancement under signal presence uncertainty using logspectral amplitude estimator,” IEEE Signal Processing Letters, vol. 9, no. 4, pp. 113–116, 2002.

[17] Y. Soon, S. N. Koh, and C. K. Yeo, “Improved noise suppression filter using selfadaptive estimator of probability of speech absence,” Signal Processing, vol. 75, no. 2, pp. 151–159, 1999.

[18] A. Abramson and I. Cohen, “Simultaneous detection and estimation approach for speech enhancement,” IEEE Transactions on Audio, Speech, and Language Processing, vol. 15, no. 8, pp. 2348–2359, 2007.

[19] M. R. Weiss, E. Aschkenasy, and T. W. Parsons, “Study and development of the intel technique for improving speech intelligibility,” tech. rep., NICOLET SCIENTIFIC CORP NORTHVALE NJ, 1975.

[20] M. Berouti, R. Schwartz, and J. Makhoul, “Enhancement of speech corrupted by acoustic noise,” in ICASSP’79. IEEE International Conference on Acoustics, Speech, and Signal Processing, vol. 4, pp. 208–211, IEEE, 1979.

[21] S. Kamath, P. Loizou, et al., “A multi-band spectral subtraction method for enhancing speech corrupted by colored noise.,” in ICASSP, vol. 4, pp. 44164–44164, Citeseer, 2002.

[22] M. T. Sadiq, N. Shabbir, and W. J. Kulesza, “Spectral subtraction for speech enhancement in modulation domain,” International Journal of Computer Science Issues (IJCSI), vol. 10, no. 4, p. 282, 2013.

[23] N. Upadhyay and A. Karmakar, “Speech enhancement using spectral subtraction-type algorithms: A comparison and simulation study,” Procedia Computer Science, vol. 54, pp. 574–584, 2015.

[24] N. Wiener, Extrapolation, interpolation, and smoothing of stationary time series, volume 2. Cambridge: MIT press, 1949.

[25] P. Scalart et al., “Speech enhancement based on a priori signal to noise estimation,” in 1996 IEEE International Conference on Acoustics, Speech, and Signal Processing Conference Proceedings, vol. 2, pp. 629–632, IEEE, 1996.

[26] J. Chen, J. Benesty, Y. Huang, and S. Doclo, “New insights into the noise reduction wiener filter,” IEEE Transactions on audio, speech, and language processing, vol. 14, no. 4, pp. 1218–1234, 2006.

[27] B. Xia and C. Bao, “Wiener filtering based speech enhancement with weighted denoising auto-encoder and noise classification,” Speech Communication, vol. 60, pp. 13–29, 2014.

[28] N. Upadhyay and R. Jaiswal, “Single channel speech enhancement: Using wiener filtering with recursive noise estimation,” Procedia Computer Science, vol. 84, pp. 22–30, 2016.

[29] M. Dendrinos, S. Bakamidis, and G. Carayannis, “Speech enhancement from noise: A regenerative approach,” Speech Communication, vol. 10, no. 1, pp. 45– 57, 1991.

[30] Y. Ephraim and H. L. V. Trees, “A signal subspace approach for speech enhancement,” IEEE transactions on acoustics, speech, and signal processing, vol. 3, no. 4, pp. 251–266, 1995.

[31] J. Huang and Y. Zhao, “A dct-based fast signal subspace technique for robust speech recognition,” IEEE transactions on acoustics, speech, and signal processing, vol. 8, no. 6, pp. 747–751, 2000.

[32] Y. Hu and P. C. Loizou, “A generalized subspace approach for enhancing speech corrupted by colored noise,” IEEE transactions on acoustics, speech, and signal processing, vol. 11, no. 4, pp. 334–341, 2003.

[33] J. Sun, C. Xie, and Y. Leng, “A signal subspace speech enhancement approach based on joint low-rank and sparse matrix decomposition,” Archives of Acoustics, vol. 41, no. 2, pp. 245–254, 2016.

[34] U. Kjems, M. S. Pedersen, J. B. Boldt, T. Lunner, and D. Wang, “Speech intelligibility of ideal binary masked mixtures,” in 2010 18th European Signal Processing Conference, pp. 1909–1913, IEEE, 2010.

[35] N. Madhu, A. Spriet, S. Jansen, R. Koning, and J. Wouters, “The potential for speech intelligibility improvement using the ideal binary mask and the ideal wiener filter in single channel noise reduction systems: Application to auditory prostheses,” IEEE Transactions on Audio, Speech, and Language Processing, vol. 21, no. 1, pp. 63–72, 2012.

[36] R. Koning, N. Madhu, and J. Wouters, “Ideal time–frequency masking algorithms lead to different speech intelligibility and quality in normal-hearing and cochlear implant listeners,” vol. 62, pp. 331–341, IEEE, 2014.

[37] T. Sreenivas and P. Kirnapure, “Codebook constrained wiener filtering for speech enhancement,” IEEE Transactions on Audio, Speech, and Language Processing, vol. 4, no. 5, pp. 383–389, 1996.

[38] S. Srinivasan, J. Samuelsson, and W. B. Kleijn, “Codebook driven short-term predictor parameter estimation for speech enhancement,” IEEE Transactions on Audio, Speech, and Language Processing, vol. 14, no. 1, pp. 163–176, 2006.

[39] S. Srinivasan, J. Samuelsson, and W. B. Kleijn, “Codebook-based bayesian speech enhancement for nonstationary environments,” IEEE Transactions on Audio, Speech, and Language Processing, vol. 15, no. 2, pp. 441–452, 2007.

[40] Q. He, C.-c. Bao, and F. Bao, “Multiplicative update of ar gains in codebookdriven speech enhancement,” in 2016 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pp. 5230–5234, IEEE, 2016.

[41] Y. Ephraim, “A bayesian estimation approach for speech enhancement using hidden markov models,” IEEE Transactions on Signal Processing, vol. 40, no. 4, pp. 725–735, 1992.

[42] D. Y. Zhao and W. B. Kleijn, “Hmm-based gain modeling for enhancement of speech in noise,” IEEE Transactions on Audio, Speech, and Language Processing, vol. 15, no. 3, pp. 882–892, 2007.

[43] H. Veisi and H. Sameti, “Speech enhancement using hidden markov models in mel-frequency domain,” Speech Communication, vol. 55, no. 2, pp. 205–220, 2013.

[44] N. Mohammadiha, R. Martin, and A. Leijon, “Spectral domain speech enhancement using hmm state-dependent super-gaussian priors,” IEEE Signal Processing Letters, vol. 20, no. 3, pp. 253–256, 2013.

[45] N. Mohammadiha and A. Leijon, “Nonnegative hmm for babble noise derived from speech hmm: Application to speech enhancement.,” IEEE Transactions on Audio, Speech and Language Processing, vol. 21, no. 5, pp. 998–1011, 2013.

[46] J. Xu, Y. Du, L. R. Dai, and C. H. Lee, “An experimental study on speech enhancement based on deep neural networks,” IEEE Signal Processing Letters, vol. 21, no. 1, pp. 65–68, 2014.

[47] J. Xu, Y. Du, L. R. Dai, and C. H. Lee, “A regression approach to speech enhancement based on deep neural networks,” IEEE/ACM Trans. Audio, Speech,Lang. Process, vol. 23, no. 1, pp. 7–19, 2015.

[48] T. T. Vu, B. Bigot, and E. S. Chng, “Combining non-negative matrix factorization and deep neural networks for speech enhancement and automatic speech recognition,” in 2016 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pp. 499–503, IEEE, 2016.

[49] M. Aharon, M. Elad, and A. Bruckstein, “K-svd: An algorithm for designing overcomplete dictionaries for sparse representation.,” IEEE Transactions on signal processing, vol. 54, no. 11, pp. 4311–4322, 2006.

[50] A. Iqbal and A. K. Seghouane, “An approach for sequential dictionary learning in nonuniform noise,” in 2017 International Conference on Digital Image Computing: Techniques and Applications (DICTA), pp. 1–5.

[51] A. K. Seghouane and A. Iqbal, “Sequential dictionary learning from correlated data: Application to fmri data analysis,” IEEE Transactions on Image Processing, vol. 26, no. 6, pp. 3002–3015, 2017.

[52] A. Iqbal and A. Seghouane, “An -divergence based approach for robust dictionary learning,” IEEE Transactions on Image Processing, vol. 28, no. 11, pp. 5729–5739, 2019.

[53] P. Stoica and R. L. Moses, Introduction to spectral analysis, volume 1. Upper Saddle River: Prentice hall, 1997.

[54] O. Cappé, “Elimination of the musical noise phenomenon with the ephraim and malah noise suppressor,” IEEE transactions on Speech and Audio Processing, vol. 2, no. 2, pp. 345–349, 1994.

[55] P. C. Yong, S. Nordholm, and H. H. Dam, “Optimization and evaluation of sigmoid function with a priori snr estimate for real-time speech enhancement,” Speech communication, vol. 55, no. 2, pp. 358–376, 2013.

[56] J. S. Lim and A. V. Oppenheim, “Enhancement and bandwidth compression of noisy speech,” Proceedings of the IEEE, vol. 67, no. 12, pp. 1586–1604, 1979.

[57] M. Portnoff, “Time-frequency representation of digital signals and systems based on shorttime fourier analysis,” IEEE Transactions on Acoustics, Speech, and Signal Processing, vol. 28, no. 1, pp. 55–69, 1980.

[58] R. Crochiere, “A weighted overlap-add method of short-time fourier analysis/ synthesis,” IEEE Transactions on Acoustics, Speech, and Signal Processing, vol. 28, no. 1, pp. 99–102, 1980.

[59] S. Nawab, T. Quatieri, and J. Lim, “Signal reconstruction from short-time fourier transform magnitude,” IEEE Transactions on Acoustics, Speech, and Signal Processing, vol. 31, no. 4, pp. 986–998, 1983.

[60] J. H. L. Hansen and B. L. Pellom, “An effective quality evaluation protocol for speech enhancement algorithms,” in Fifth international conference on spoken language processing, pp. 2819–2822, 1998.

[61] D. Klatt, “Prediction of perceived phonetic distance from critical-band spectra: A first step,” in ICASSP’82. IEEE International Conference on Acoustics, Speech, and Signal Processing, vol. 7, pp. 1278–1281, IEEE, 1982.

[62] Y. Hu and P. C. Loizou, “Evaluation of objective quality measures for speech enhancement,” IEEE Transactions on audio, speech, and language processing, vol. 16, no. 1, pp. 229–238, 2007.

[63] S. R. Quackenbush, “Objective measures of speech quality (subjective).,” Ph.D. dissertation, 1986.

[64] H. Veisi and H. Sameti, “Hidden-markov-model-based voice activity detector with high speech detection rate for speech enhancement,” IET signal processing, vol. 6, no. 1, pp. 54–63, 2012.

[65] H. Wei, Y. Long, and H. Mao, “Improvements on self-adaptive voice activity detector for telephone data,” International Journal of Speech Technology, vol. 19, no. 3, pp. 623–630, 2016.

[66] S. Boll, “Suppression of acoustic noise in speech using spectral subtraction,” IEEE Transactions on acoustics, speech, and signal processing, vol. 27, no. 2, pp. 113–120, 1979.

[67] M. A. Abd El-Fattah, M. I. Dessouky, A. M. Abbas, S. M. Diab, E.-S. M. El-Rabaie, W. Al-Nuaimy, S. A. Alshebeili, and F. E. Abd El-Samie, “Speech enhancement with an adaptive wiener filter,” International Journal of Speech Technology, vol. 17, no. 1, pp. 53–64, 2014.

[68] C. H. You, S. N. Koh, and S. Rahardja, “/spl beta/-order mmse spectral amplitude estimation for speech enhancement,” IEEE transactions on speech and audio processing, vol. 13, no. 4, pp. 475–486, 2005.

[69] I. T. Jolliffe and J. Cadima, “Principal component analysis: a review and recent developments,” Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences, vol. 374, no. 2065, p. 20150202, 2016.

[70] I. Koch, Analysis of multivariate and high-dimensional data, vol. 32. Cambridge University Press, 2013.

[71] S. Pyatykh, J. Hesser, and L. Zheng, “Image noise level estimation by principal component analysis,” IEEE transactions on image processing, vol. 22, no. 2, pp. 687–699, 2012.

[72] L. Zhang, R. Lukac, X. Wu, and D. Zhang, “Pca-based spatially adaptive denoising of cfa images for single-sensor digital cameras,” IEEE transactions on image processing, vol. 18, no. 4, pp. 797–812, 2009.

[73] H. Khalilian and I. V. Bajic, “Video watermarking with empirical pca-based decoding,” IEEE transactions on image processing, vol. 22, no. 12, pp. 4825– 4840, 2013.

[74] N. Vaswani and R. Chellappa, “Principal components null space analysis for image and video classification,” IEEE Transactions on Image Processing, vol. 15, no. 7, pp. 1816–1830, 2006.

[75] A. H. Andersen, D. M. Gash, and M. J. Avison, “Principal component analysis of the dynamic response measured by fmri: a generalized linear systems framework,” Magnetic resonance imaging, vol. 17, no. 6, pp. 795–815, 1999.

[76] J. Yang, D. Zhang, A. F. Frangi, and J.-y. Yang, “Two-dimensional pca: a new approach to appearance-based face representation and recognition,” IEEE transactions on pattern analysis and machine intelligence, vol. 26, no. 1, pp. 131–137, 2004.

[77] C. M. Bishop, Pattern recognition and machine learning. springer, 2006.

[78] Y. Hu, “Subjective evaluation and comparison of speech enhancement algorithms,” Speech Communication, vol. 49, pp. 588–601, 2007.

[79] J. Du and Q. Huo, “A speech enhancement approach using piecewise linear approximation of an explicit model of environmental distortions,” in Ninth annual conference of the international speech communication association, 2008.

[80] Alsheibi, A. Z., Valavanis, K. P., Iqbal, A., & Aman, M. N. (2022, May). Speech Enhancement Framework with Noise Suppression Using Block Principal Component Analysis. In Acoustics (Vol. 4, No. 2, pp. 441-459). MDPI.