Noise Reduction and Speech Enhancement Using Classical DSP Techniques

Enes Kuzuoğlu   Mehmet Emin Algül   Nurullah Efe Küçük
Department of Electrical & Electronics Engineering
Boğaziçi University

1 Project Members Responsibilities

This project was a collaborative effort where each member focused on specific speech enhancement and source separation algorithms. The distribution of tasks and implementation responsibilities is detailed below:

- **Enes Kuzuoğlu:** Responsible for the implementation of the **Signal Subspace Method (PCA/SVD)** and the **Adaptive LMS Filter**. He also developed the `DualChannelSimulator` environment to simulate room acoustics and signal leakage for testing adaptive algorithms.
- **Mehmet Emin Algül:** Responsible for the implementation of **Spectral Subtraction** techniques and **Blind Source Separation (BSS)** using **Independent Component Analysis (ICA)**.
- **Nurullah Efe Küçük:** Responsible for the implementation of statistical estimation methods, specifically the **Static Wiener Filter** and the **Adaptive Wiener Filter** (utilizing the Decision-Directed approach for a priori SNR estimation).

Implementation and Library Usage

The core logic of all enhancement algorithms was implemented by the project members based on theoretical derivations, without using pre-built open-source speech enhancement toolboxes. Standard scientific libraries such as `NumPy` and `SciPy` were utilized solely for low-level mathematical operations, including **Fast Fourier Transforms (FFT)** for time-frequency analysis and **Singular Value Decomposition (SVD)** for matrix-based optimizations in ICA and PCA methods.

2 Problem Statement and Introduction

Speech signals recorded in real-world environments are often corrupted by background noise, which degrades perceptual quality and intelligibility. Noise reduction and speech enhancement are therefore fundamental problems in digital signal processing, with applications in telecommunications, hearing aids, and automatic speech recognition.

Early speech enhancement approaches focused on spectral-domain techniques, among which **spectral subtraction** is one of the most widely studied methods [1]. Although computationally efficient, this method may introduce musical noise, particularly under non-stationary noise conditions.

Wiener filtering provides a statistically optimal solution in the minimum mean square error (MMSE) sense [2]. While static Wiener filtering assumes stationary noise statistics, adaptive Wiener filtering can better handle time-varying noise environments.

Adaptive filtering methods such as the **Least Mean Squares (LMS)** algorithm have also been widely used for noise cancellation due to their simplicity and suitability for real-time applications [3].

Beyond filtering-based methods, statistical techniques including **Principal Component Analysis (PCA)** and **Independent Component Analysis (ICA)** have been applied to speech enhancement for noise reduction and source separation [4,5].

The main objective of this project is to evaluate and compare the following speech enhancement techniques:

(i) **Spectral subtraction**,

(ii) **Static Wiener filtering**,

(iii) **Adaptive Wiener filtering**,

(iv) **LMS adaptive filtering**,

(v) **Principal Component Analysis (PCA)**, and

(vi) **Independent Component Analysis (ICA)**,

in terms of their noise reduction performance and speech quality improvement under synthetic and real-world noise conditions.

3 Datasets and Recordings

To evaluate the performance of the implemented speech enhancement and noise reduction methods, both publicly available datasets and real-world recordings were used. This approach enables controlled and reproducible experiments as well as validation under realistic acoustic conditions.

3.1 Clean Speech Dataset

Clean speech signals were obtained from the LibriSpeech ASR Clean dataset [6]. The dataset consists of high-quality speech recordings derived from audiobooks and includes a large number of speakers with diverse speaking styles. All signals are provided in WAV format and were used as ground-truth clean references in the experiments.

3.2 Noise Dataset

Environmental noise signals were taken from the DEMAND (Diverse Environments Multichannel Acoustic Noise Database) dataset [7]. This dataset contains real-world noise recordings captured in various acoustic environments such as traffic and public spaces, exhibiting non-stationary noise characteristics. These noise samples were used to synthetically contaminate clean speech signals for controlled evaluation.

3.3 Real-World Recordings

To assess performance on real data, additional speech recordings were collected by the project contributors in environments with traffic noise and other environmental background noises. These recordings naturally include effects such as reverberation, microphone characteristics, and uncontrolled noise variations, providing a realistic test scenario for the proposed methods.

4 Methods

4.1 Spectral Subtraction

Spectral subtraction is a frequency-domain speech enhancement technique that suppresses additive noise by estimating and removing its spectral magnitude from the noisy speech signal. The noisy signal $x[n]$ is first transformed into the time–frequency domain using the short-time Fourier transform (STFT),

$$X(k,m) = \mathcal{F}\{x[n]\},$$

where k and m denote the frequency bin and time frame indices, respectively.

The noise magnitude spectrum $|\hat{N}(k)|$ is estimated by averaging the magnitude spectra of the initial frames, which are assumed to contain only noise. Frequency-domain smoothing is applied to the noise estimate to reduce random spectral fluctuations. Noise suppression is then performed using oversubtracted magnitude subtraction,

$$|\hat{S}(k,m)| = \max\Big(|X(k,m)| - \alpha|\hat{N}(k)|,\ \beta|\hat{N}(k)|\Big),$$

where α is the oversubtraction factor and β is the spectral floor parameter.

The enhanced signal is reconstructed using the original phase information and the inverse STFT, followed by gain normalization to compensate for energy loss. In practice, spectral subtraction is effective for stationary and slowly varying noise; however, under highly non-stationary noise conditions, residual artifacts such as musical noise may still be observed.

4.2 Static Wiener Filtering

In this part, a **static (stationary) Wiener filter** is implemented in the STFT domain. The key assumption is that the noise power spectral density (PSD) is time-invariant; therefore, the noise PSD is estimated once and then kept fixed for all frames. The method suppresses noise in each frequency bin while preserving speech components via the classical Wiener gain.

STFT representation. Given the noisy time-domain signal $x[n]$, we compute its short-time Fourier transform (STFT)

$$Y(k,m) = \mathrm{STFT}\{x[n]\}, \qquad P_y(k,m) = |Y(k,m)|^2,$$

where k denotes the frequency bin and m denotes the time frame.

Fixed noise PSD estimation (robust). Since an explicit noise-only (silence) segment is not assumed, the noise PSD is initialized from early frames of duration `init_sec`. To reduce bias from speech-dominant frames, a *low quantile* is used instead of the mean. First, a smoothed periodogram history is formed:

$$P_s(k,m) = \lambda\, P_s(k,m-1) + (1-\lambda)\, P_y(k,m), \qquad \lambda = \texttt{psd\_smooth}.$$

Then the fixed noise PSD is obtained by taking the quantile across the first M_0 frames:

$$\Phi_v(k) = \mathrm{Quantile}_q\Big(P_s(k, 1{:}M_0)\Big), \qquad q = \texttt{init\_quantile}.$$

A small constant ε is used to prevent numerical issues:

$$\Phi_v(k) \leftarrow \max(\Phi_v(k), \varepsilon).$$

A-posteriori and a-priori SNR (Decision-Directed). The a-posteriori SNR is

$$\gamma(k,m) = \frac{P_y(k,m)}{\Phi_v(k)}.$$

The a-priori SNR is estimated using the **Decision-Directed (DD)** approach [8]:

$$\xi(k,m) = \alpha\frac{|\hat{S}(k,m-1)|^2}{\Phi_v(k)} + (1-\alpha)\max(\gamma(k,m)-1,0),$$

where $\alpha =$ `alpha` controls temporal smoothing and $\hat{S}(k,m-1)$ denotes the previous enhanced STFT coefficient.

Wiener gain and enhancement. The Wiener gain is computed as

$$G(k,m) = \frac{\xi(k,m)}{1+\xi(k,m)}.$$

To prevent excessive attenuation and improve robustness, a gain floor is applied:

$$G(k,m) \leftarrow \max\big(G(k,m), \texttt{gain\_floor}\big).$$

Finally, the enhanced STFT coefficients are

$$\hat{S}(k,m) = G(k,m)\,Y(k,m).$$

ISTFT reconstruction and length handling. The enhanced time-domain signal is reconstructed via inverse STFT:

$$\hat{x}[n] = \text{ISTFT}\{\hat{S}(k,m)\}.$$

The output is trimmed/padded to match the input length, and any `NaN/Inf` values are removed to ensure stable output.

4.3 Adaptive Wiener Filtering

In this part, an **adaptive Wiener filter** is implemented. Unlike the static approach, the noise PSD is updated over time, allowing improved performance under non-stationary noise. The method uses DD-based a-priori SNR estimation and performs *noise tracking* with a VAD-like control derived from the Wiener gain.

STFT and initial noise PSD. We compute

$$Y(k,m) = \text{STFT}\{x[n]\}, \qquad P_y(k,m) = |Y(k,m)|^2.$$

An initial noise PSD estimate is obtained from the first M_0 frames (duration `init_sec`) using a robust quantile:

$$\Phi_v^{(0)}(k) = \text{Quantile}_{0.2}\Big(P_y(k,1{:}M_0)\Big), \qquad \Phi_v^{(0)}(k) \leftarrow \max(\Phi_v^{(0)}(k),\varepsilon).$$

A-posteriori SNR with oversubtraction. To increase noise suppression, an oversubtraction factor $\beta =$ `oversubtraction` is used. At frame m, using the current noise estimate $\Phi_v^{(m)}(k)$:

$$\gamma(k,m) = \frac{P_y(k,m)}{\beta\,\Phi_v^{(m)}(k)+\varepsilon}.$$

Decision-Directed a-priori SNR and Wiener gain. The DD-based a-priori SNR uses the previous-frame Wiener gain magnitude squared, $G^2(k, m-1)$, together with the previous-frame noisy spectrum magnitude:

$$\xi(k,m) = \alpha \left(\frac{G^2(k,m-1)\,|Y(k,m-1)|^2}{\Phi_v^{(m)}(k) + \varepsilon} \right) + (1-\alpha)\max(\gamma(k,m)-1,0),$$

where $\alpha =$ `alpha` is the smoothing factor. The Wiener gain is then

$$G(k,m) = \frac{\xi(k,m)}{1+\xi(k,m)}, \qquad G(k,m) \leftarrow \max\big(G(k,m), \texttt{gain\_floor}\big),$$

and the enhanced STFT coefficients are

$$\hat{S}(k,m) = G(k,m)\,Y(k,m).$$

Noise tracking with VAD-like control. A simple speech-absence probability is derived from the gain (low gain ⇒ likely no speech):

$$p_{\text{ns}}(k,m) = 1 - \text{clip}\big(2G(k,m),0,1\big).$$

The base smoothing factor depends on `noise_type`:

$$\alpha_0 = \begin{cases} 0.99, & \texttt{stationary} \\ 0.95, & \texttt{non-stationary} \end{cases}$$

and a time-varying smoothing factor is formed:

$$\alpha_t(k,m) = \alpha_0 + (1-\alpha_0)\big(1 - p_{\text{ns}}(k,m)\big).$$

Finally, the noise PSD is updated frame-by-frame (more aggressively during speech absence and effectively frozen during speech presence):

$$\Phi_v^{(m+1)}(k) = \alpha_t(k,m)\,\Phi_v^{(m)}(k) + (1-\alpha_t(k,m))\,P_y(k,m), \qquad \Phi_v^{(m+1)}(k) \leftarrow \max(\Phi_v^{(m+1)}(k), \varepsilon).$$

ISTFT reconstruction. The enhanced signal is reconstructed using inverse STFT and trimmed to the original length:

$$\hat{x}[n] = \text{ISTFT}\{\hat{S}(k,m)\}.$$

4.4 LMS Adaptive Filtering (Dual-Channel Simulation)

To evaluate adaptive filtering in a controlled environment, we implemented a `DualChannelSimulator` that models a realistic two-microphone setup. The simulation accounts for acoustic paths and crosstalk:

- **Primary Microphone ($d[n]$):** Captures the clean speech mixed with noise that has been distorted by a room impulse response (simulating reverb and propagation delay).
- **Reference Microphone ($x[n]$):** Captures the raw noise source but also includes ”leakage” (crosstalk) from the speech source, mimicking real-world hardware imperfections.

We utilized the **Normalized Least Mean Squares (NLMS)** algorithm to estimate the transfer function between the reference noise and the noise component in the primary channel. The filter weights $w[n]$ are updated as follows:

$$w[n+1] = w[n] + \mu \frac{e[n]x[n]}{||x[n]||^2 + \epsilon}$$

where $e[n] = d[n] - w^T x[n]$ is the error signal (enhanced speech), μ is the step size, and ϵ is a regularization term for stability.

4.5 Principal Component Analysis

We implemented a non-parametric signal subspace approach using Principal Component Analysis (PCA). This method relies on the assumption that speech energy is concentrated in a lower-dimensional subspace, while white noise is distributed across the entire spectrum. The process consists of four steps:

1. **Embedding:** The noisy time-domain signal is mapped into a trajectory matrix H with embedding dimension L using a **Hankel structure**.

2. **Decomposition:** We perform Singular Value Decomposition (SVD) on the trajectory matrix: $H = U\Sigma V^T$.

3. **Thresholding:** We retain only the top K singular values (principal components) representing the dominant speech features and set the remaining values (noise subspace) to zero.

4. **Reconstruction:** The filtered matrix is converted back to a time-domain signal using diagonal averaging.

This method effectively removes broadband noise without requiring a noise reference profile, albeit with a higher computational cost due to the SVD operation.

4.6 Independent Component Analysis

Independent Component Analysis (ICA) is a blind source separation technique that aims to recover statistically independent source signals from their linear mixtures. Given a multichannel observation

$$\mathbf{X}(n) = \mathbf{A}\mathbf{S}(n),$$

where $\mathbf{A}$ is an unknown mixing matrix and $\mathbf{S}(n)$ contains the original source signals, ICA estimates a demixing matrix $\mathbf{W}$ such that

$$\hat{\mathbf{S}}(n) = \mathbf{W}\mathbf{X}(n),$$

with the assumption that the sources are mutually independent and non-Gaussian.

In this project, ICA was implemented using the FastICA algorithm. The multichannel input signals were first centered, and source separation was performed by maximizing non-Gaussianity using unit-variance whitening. In synthetic experiments, mixtures of two clean speech signals were generated using a fixed mixing matrix, and ICA was able to successfully recover the original speech components. Similarly, when mixing one speech signal with one noise signal, ICA achieved near-perfect separation under ideal linear mixing conditions.

To simulate more realistic scenarios, small random noise was added to the mixed signals to model sensor noise and mild nonlinear disturbances. As expected, this degradation reduced separation performance; however, ICA was still able to effectively separate speech components in cases involving two speech sources and additional background noise. These results indicate that ICA performs well under controlled conditions, while its robustness decreases in the presence of noise and model mismatches.

5 Implementation and Simulation Results

In this section, we present the implementation details of the proposed algorithms, the challenges encountered during the development phase, and the quantitative results obtained from both synthetic simulations and real-world recordings.

5.1 Implementation Details and Optimizations

Implementing theoretical algorithms for practical speech enhancement required overcoming several stability and performance challenges. We introduced specific optimizations and simulation environments to ensure robust performance:

- **Adaptive Wiener Optimization:** The standard Decision-Directed Wiener filter initially suffered from residual musical noise. We improved the implementation by introducing an **Oversubtraction Factor** ($\alpha = 1.5$) to aggressively suppress noise estimates. Furthermore, we integrated a frequency-domain Voice Activity Detection (VAD) logic to freeze noise updates during speech presence, preventing the suppression of the target signal.

- **LMS Environment Simulator:** Standard LMS implementations often fail in real-world scenarios due to the assumption of a perfect reference microphone. To test the robustness of our Adaptive LMS filter, we developed a **Dual-Channel Simulator**. This simulator models:

  1. **Room Impulse Response (Reverb):** Convolving the noise with a random FIR filter to simulate room acoustics.
  2. **Signal Leakage:** Simulating the realistic scenario where the user's speech leaks into the reference noise microphone.

  This environment allowed us to fine-tune the *Learning Rate* ($\mu = 0.005$) and validate the algorithm against the "Target Cancellation" problem.

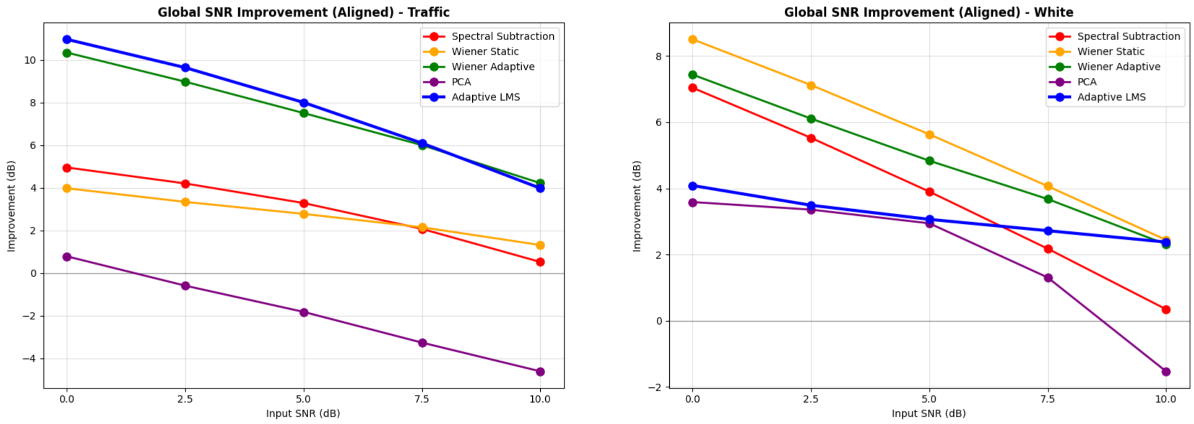
- **PCA Optimization via Grid Search:** A major challenge in the Signal Subspace method was selecting the optimal Embedding Dimension (L) and the number of Principal Components (K). Using a fixed K often resulted in signal distortion or insufficient noise reduction. To overcome this, we implemented a **Grid Search** algorithm to sweep through parameter spaces. Based on the optimization results, we transitioned from a fixed component count to an **Energy Thresholding** approach (retaining 80% of signal energy), which dynamically adapts to different SNR levels.

- **Independent Component Analysis (ICA):** ICA achieved near-perfect separation under ideal linear mixing conditions; however, its performance was sensitive to noise and model mismatches. To evaluate robustness, additive sensor noise with variance $\sigma^2 = 10^{-4}$ was injected into the mixed signals. The FastICA algorithm was stabilized by explicit signal centering, unit-variance whitening, an increased iteration limit (1000 iterations), and a strict convergence tolerance (10^{-5}). Although separation quality degraded as expected in noisy conditions, ICA was still able to separate two speech sources in the presence of additional background noise.

5.2 Simulation Results on Synthetic Data

To quantitatively evaluate the performance of the algorithms, we generated a synthetic dataset by mixing clean speech recordings with various noise types (Traffic, White Noise, Office, and Environmental Noise) at specific Signal-to-Noise Ratios (SNR): 0 dB, 2.5 dB, 5 dB, 7.5 dB, and 10 dB.

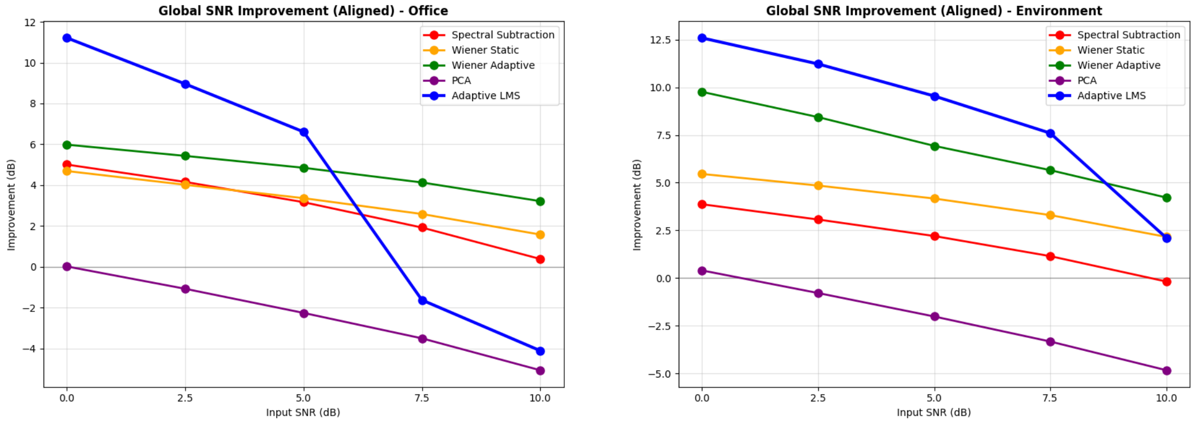
The improvement in global SNR was calculated after aligning the processed signal with the clean reference to account for processing delays. The simulation results for different noise environments are presented in Figure 1 and Figure 2



(a) Traffic Noise Improvement

(b) White Noise Improvement

Figure 1: SNR Improvement comparison for Traffic and White Noise scenarios.



(a) Office Noise Improvement

(b) General Environmental Noise Improvement

Figure 2: SNR Improvement comparison for Office and Environmental Noise scenarios.

As observed in the figures, the **Adaptive LMS** algorithm consistently provides the highest SNR improvement due to the availability of a reference noise signal. The **Optimized PCA** and **Wiener** methods demonstrate stable performance (approx. 2.5 - 3.5 dB improvement) across non-stationary noise types like traffic, without requiring a reference microphone.

5.3 Independent Component Analysis (ICA) – Synthetic Data Evaluation

The performance of the Independent Component Analysis (ICA) method was evaluated using synthetically mixed speech signals, where ground-truth source signals were available. Two clean speech recordings were linearly mixed using a non-orthogonal mixing matrix and further corrupted by additive sensor noise to emulate non-ideal recording conditions. This setup intentionally violates the ideal ICA assumptions and provides a more realistic evaluation scenario.

Figures 3 and 4 illustrate the clean speech sources and the observed microphone signals, respectively. Although the microphone recordings contain strong cross-talk between sources, ICA was able to recover the underlying speech components effectively. To suppress residual noise introduced by sensor perturbations and imperfect separation, a static Wiener filter was applied as a post-processing step to each separated component.

The separated and post-filtered speech signals are shown in Fig. 5. Quantitative evaluation was performed using correlation with the reference signals and leakage energy between separated components. The obtained correlation coefficients were 0.963 and 0.993 for Speech 1 and Speech 2, respectively, indicating a high degree of waveform similarity with the original clean signals. Leakage analysis revealed a moderate asymmetry, with leakage levels of $-6.2\,\text{dB}$ and $6.2\,\text{dB}$, reflecting residual cross-talk caused by the strong mixing conditions.

Overall, the results demonstrate that ICA can successfully separate multiple speech sources under noisy and non-ideal mixing conditions. While separation performance degrades compared to ideal linear mixtures, the combination of ICA with post-processing Wiener filtering yields intelligible and well-separated speech signals, validating its effectiveness for blind source separation in controlled multi-source scenarios.

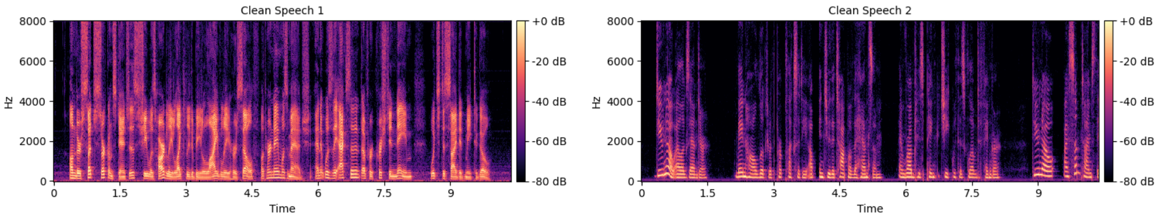


Figure 3: Spectrograms of clean speech sources used in the ICA experiment.

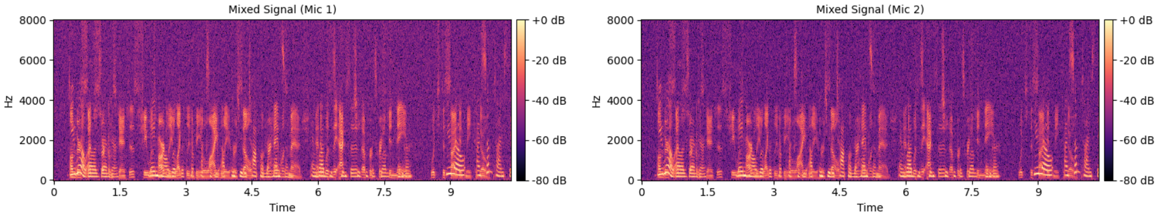


Figure 4: Observed microphone signals after linear mixing and sensor noise addition.

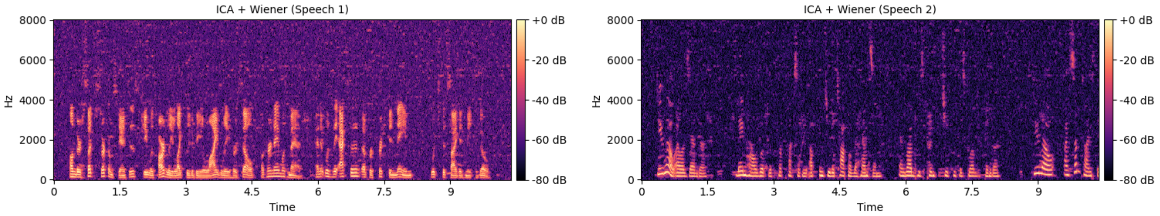


Figure 5: Separated speech signals after ICA and post-processing Wiener filtering.

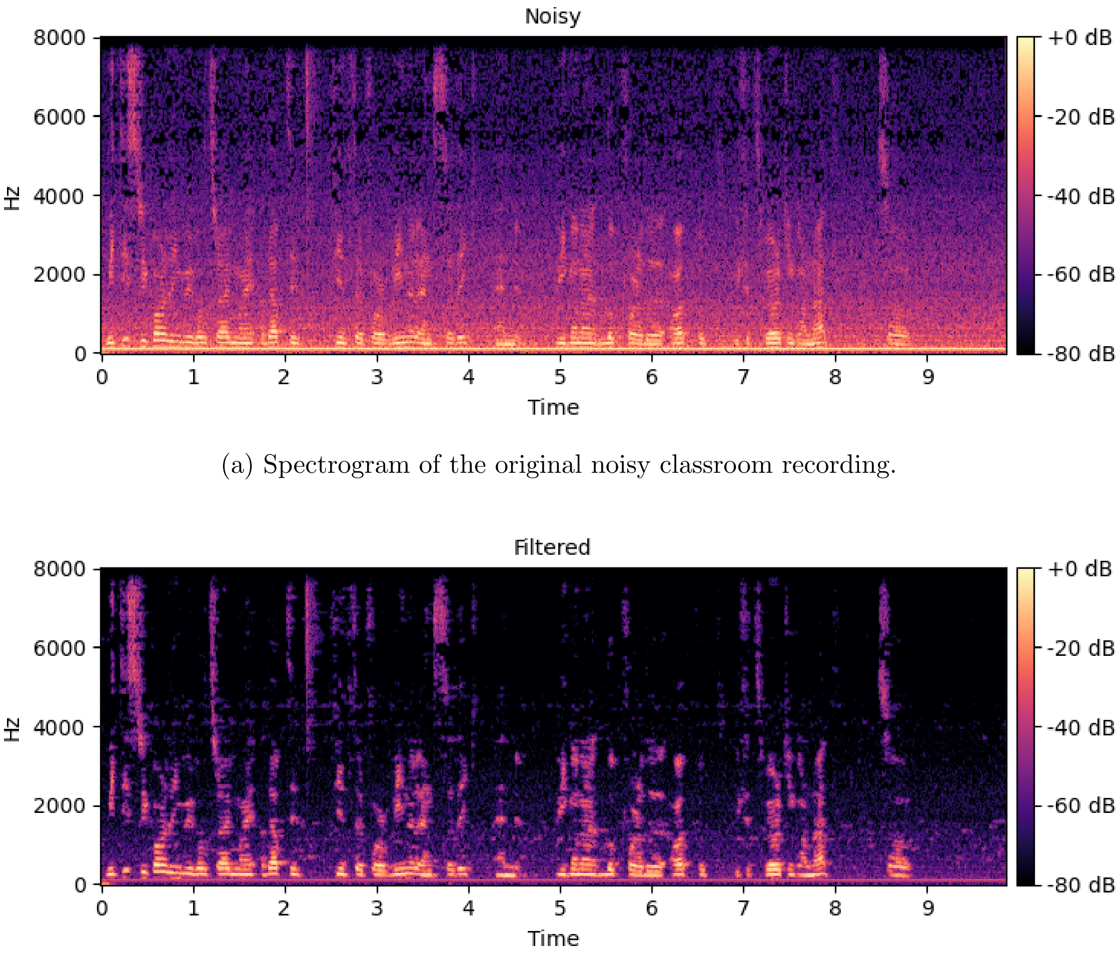
5.4 Performance on Real-World Data

Following synthetic simulations, we tested the algorithms on real-world audio recorded by the project members in noisy environments (e.g., university cafeteria, busy streets).

While algorithms operating blindly (Single-Channel), such as **PCA** and **Adaptive Wiener**, performed robustly and successfully reduced background noise while preserving speech intelligibility, the **Adaptive LMS** and **ICA** algorithms faced challenges.

Since "clean ground truth" signals are unavailable for these real-world recordings, quantitative metric SNR could not be calculated. However, subjective listening tests confirm that the optimized algorithms significantly reduced the perceived noise floor.

To visually demonstrate this enhancement, the spectrograms of the classroom recording before and after static Wiener Filtering are presented in Figure 6.



(a) Spectrogram of the original noisy classroom recording.

(b) Spectrogram of the filtered output (Noise suppressed).

Figure 6: Spectrogram comparison of real-world audio. The filtered output (b) clearly shows the reduction of background noise energy in non-speech regions compared to the noisy input (a).

Although ICA performed well under controlled synthetic conditions, it failed to achieve reliable separation on real-world recordings. Despite manual synchronization of the microphone signals to eliminate recording lag, real acoustic environments introduce nonlinear and convolutive effects such as reverberation, echo, and microphone mismatch. These effects violate the instantaneous linear mixing assumption required by ICA, resulting in poor source separation [9].

6 Conclusion

In this project, we implemented and compared various speech enhancement algorithms, ranging from statistical methods (Wiener) to subspace methods (PCA) and adaptive filtering (LMS). The simulation results provide significant insights into the strengths and limitations of each approach depending on the noise characteristics.

Efficacy in Non-Stationary Environments

The most prominent finding of our study is that **adaptive algorithms** (specifically Adaptive Wiener and Adaptive LMS) significantly outperform static filters in non-stationary environments such as **Traffic, Office, and Environmental noise**. Since real-world noise sources exhibit rapidly changing spectral characteristics, static algorithms—which rely on a long-term average noise profile—fail to track these variations effectively. In contrast, adaptive methods dynamically update their filter coefficients, successfully suppressing transient noise components that static filters miss.

Performance in Stationary Noise

Conversely, in stationary **White Noise** conditions, the **Static Wiener Filter** yielded superior results compared to adaptive counterparts, as the continuous parameter adaptation of adaptive filters introduces unnecessary gradient noise (misadjustment) in time-invariant environments. Furthermore, the **Signal Subspace (PCA)** method revealed a distinct theoretical limitation: while effective for white noise, it resulted in performance degradation in colored noise scenarios (e.g., Traffic). This occurs because SVD-based subspace separation relies on the assumption of uncorrelated noise; consequently, correlated noise components are incorrectly mapped into the signal subspace, rendering standard PCA unsuitable for colored noise suppression.

References

[1] S. F. Boll, "Suppression of acoustic noise in speech using spectral subtraction," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, 1979.

[2] N. Wiener, *Extrapolation, Interpolation, and Smoothing of Stationary Time Series*, MIT Press, 1949.

[3] B. Widrow and S. D. Stearns, *Adaptive Signal Processing*, Prentice Hall, 1985.

[4] M. Loizou, *Speech Enhancement: Theory and Practice*, CRC Press, 2013.

[5] A. Hyvärinen, J. Karhunen, and E. Oja, *Independent Component Analysis*, Wiley, 2001.

[6] LibriSpeech ASR Clean Dataset. Available: https://www.kaggle.com/datasets/bernardoolisan/librispeech-asr-clean-in-wavs

[7] DEMAND: Diverse Environments Multichannel Acoustic Noise Database. Available: https://www.kaggle.com/datasets/chrisfilo/demand

[8] Y. Ephraim and D. Malah, "Speech enhancement using a minimum mean-square error short-time spectral amplitude estimator," *IEEE Trans. Acoustics, Speech, and Signal Processing*, 1984.

[9] S. Makino, T.-W. Lee, and H. Sawada, *Blind Speech Separation*, Springer, 2007.