Voice Signal Processing Project in MATLAB

Author: safa bazrafshan

Date: April 19, 2025

Email: safa.bazrafshan@gmail.com

1. Introduction

Voice signal processing is a key area in audio engineering and telecommunications. This project simulates a basic voice signal processing workflow in MATLAB. It generates a clean voice signal, adds noise, applies a filter to remove the noise, and compares the results in both time and frequency domains.

2. Project Objectives

- Understand signal processing in MATLAB

- Apply digital filtering to remove noise

- Analyze signals in time and frequency domains

- Visualize results using charts

3. Methodology

3.1. Clean Signal Generation: A clean signal was created by combining two sinusoidal waves with frequencies of 300 Hz and 600 Hz.

3.2. Adding Noise: Gaussian white noise with an amplitude of 0.2 was added to simulate real-world environmental noise.

3.3. Filtering: A 4th-order Butterworth low-pass filter with a 1000 Hz cutoff frequency was applied to remove high-frequency noise.

3.4. Analysis: The clean, noisy, and filtered signals were compared in the time and frequency domains using plots.

4. Results

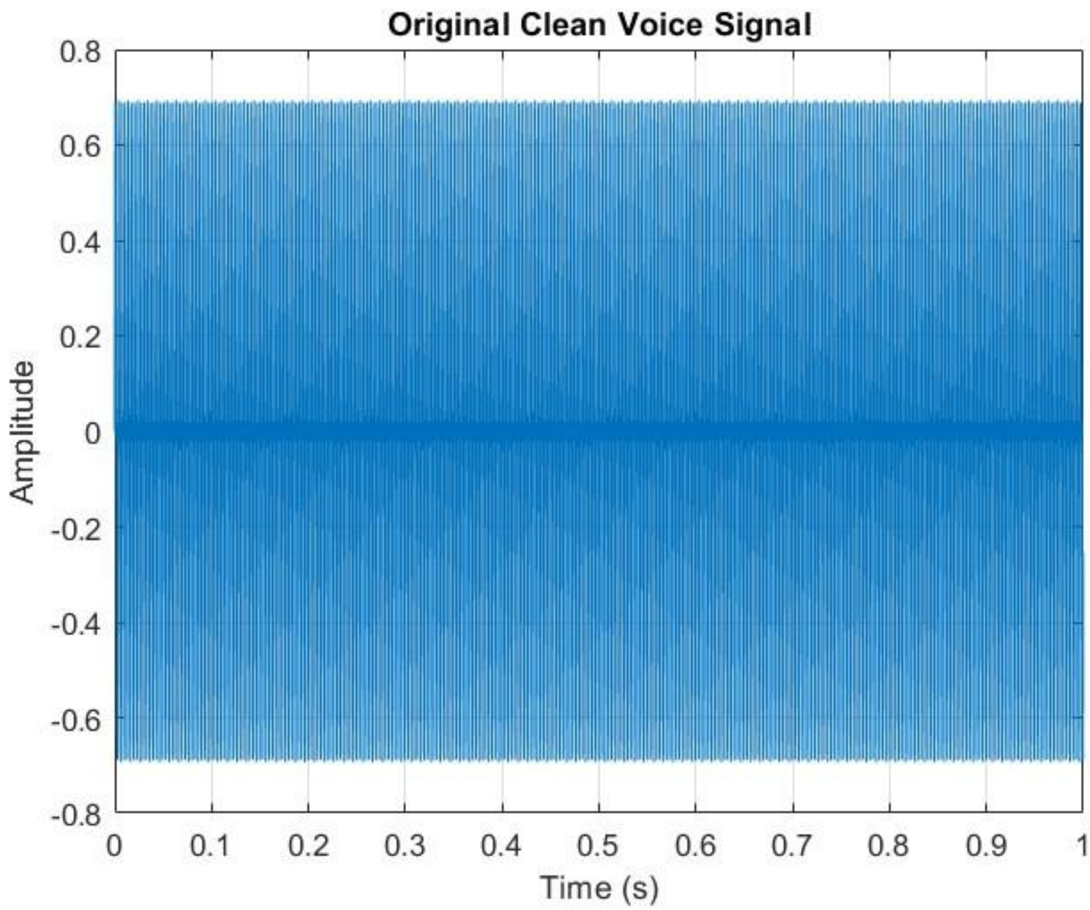

Figure 1: Time-domain comparison of clean, noisy, and filtered signals.

The clean signal shows a regular sinusoidal pattern, the noisy signal exhibits random fluctuations due to noise, and the filtered signal closely resembles the clean signal with reduced noise.

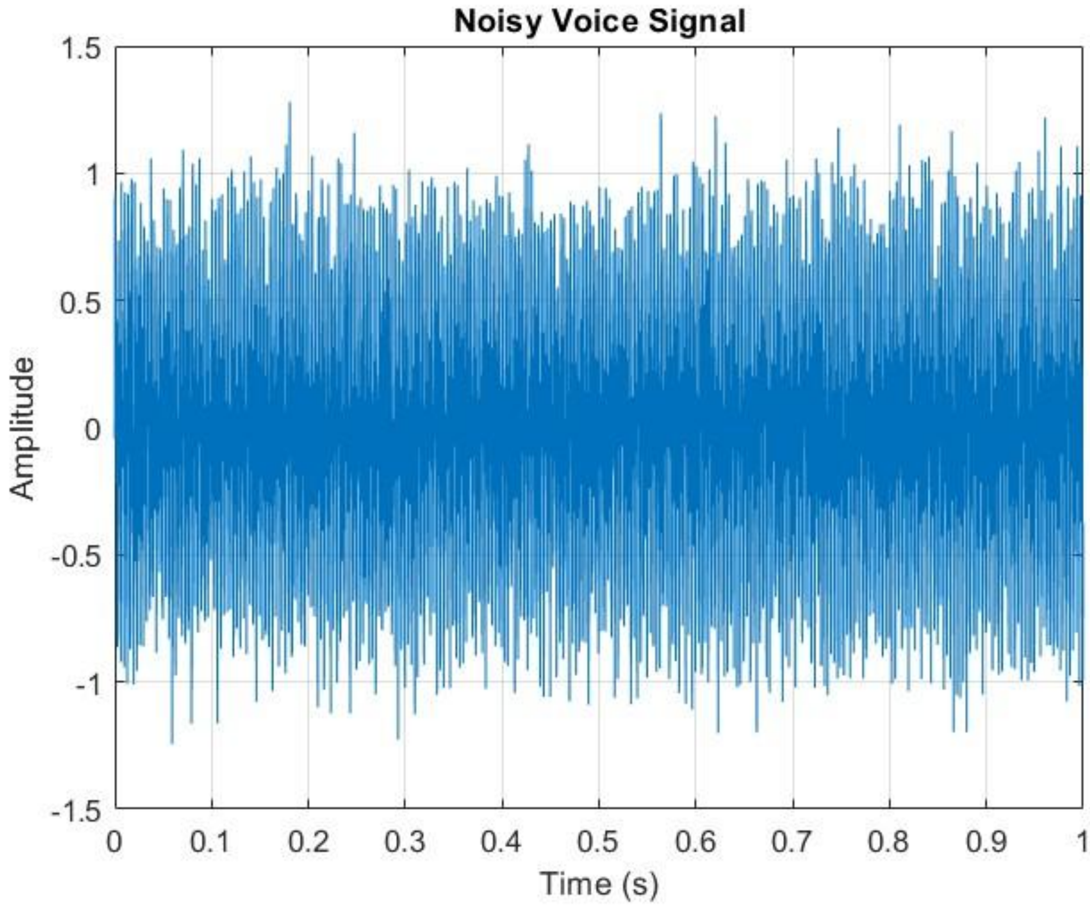

Figure 2: Frequency-domain comparison of clean, noisy, and filtered signals.

The clean signal shows clear peaks at 300 Hz and 600 Hz. The noisy signal has similar peaks but with noise across all frequencies. The filtered signal shows reduced noise in higher frequencies, closely matching the clean signal.

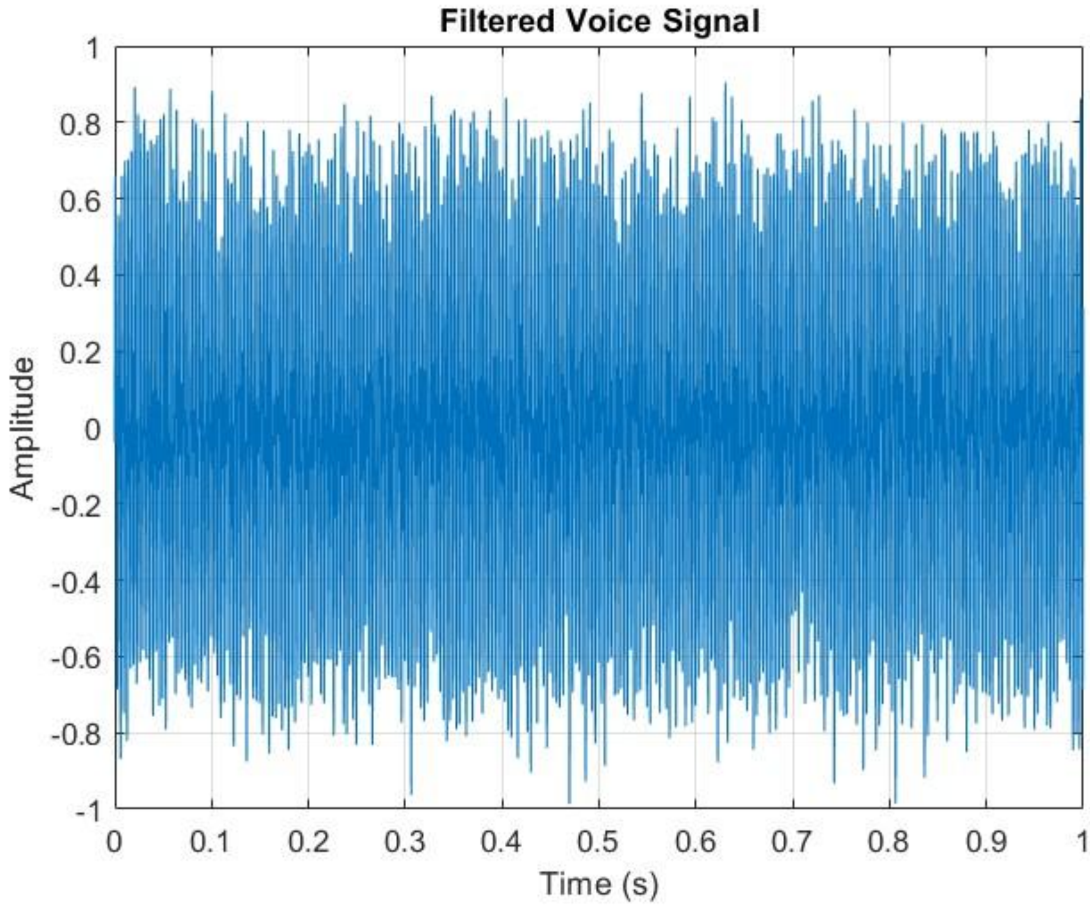

Figure 3: Filtering the signal to remove noise: To remove noise, a low-pass filter is used to eliminate high frequencies (which are mainly noise).

Low-pass filter: A Butterworth type filter with a cutoff frequency of 1000 Hz is designed. This frequency is chosen to preserve the main signal frequencies (300 and 600 Hz) while eliminating high-frequency noise.

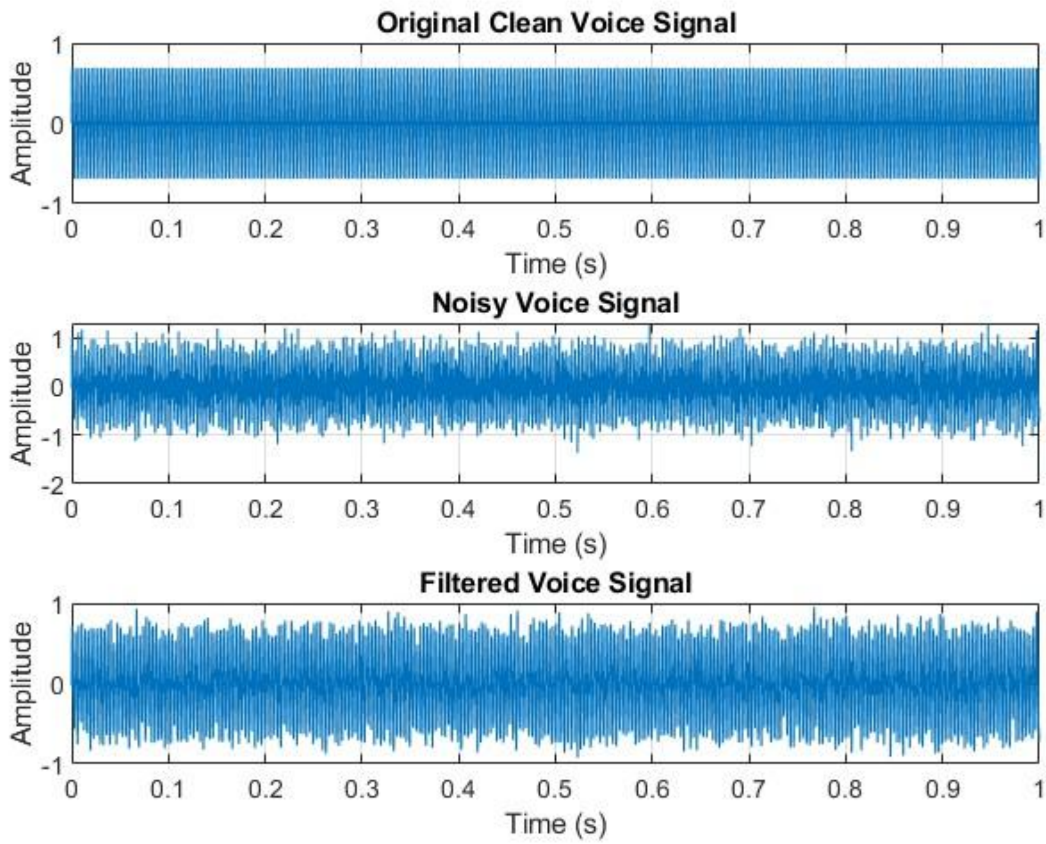

Figure 4: Comparison in the time domain.

Three plots are drawn in a single figure using subplots to compare the clean, noisy, and filtered signals in the time domain.

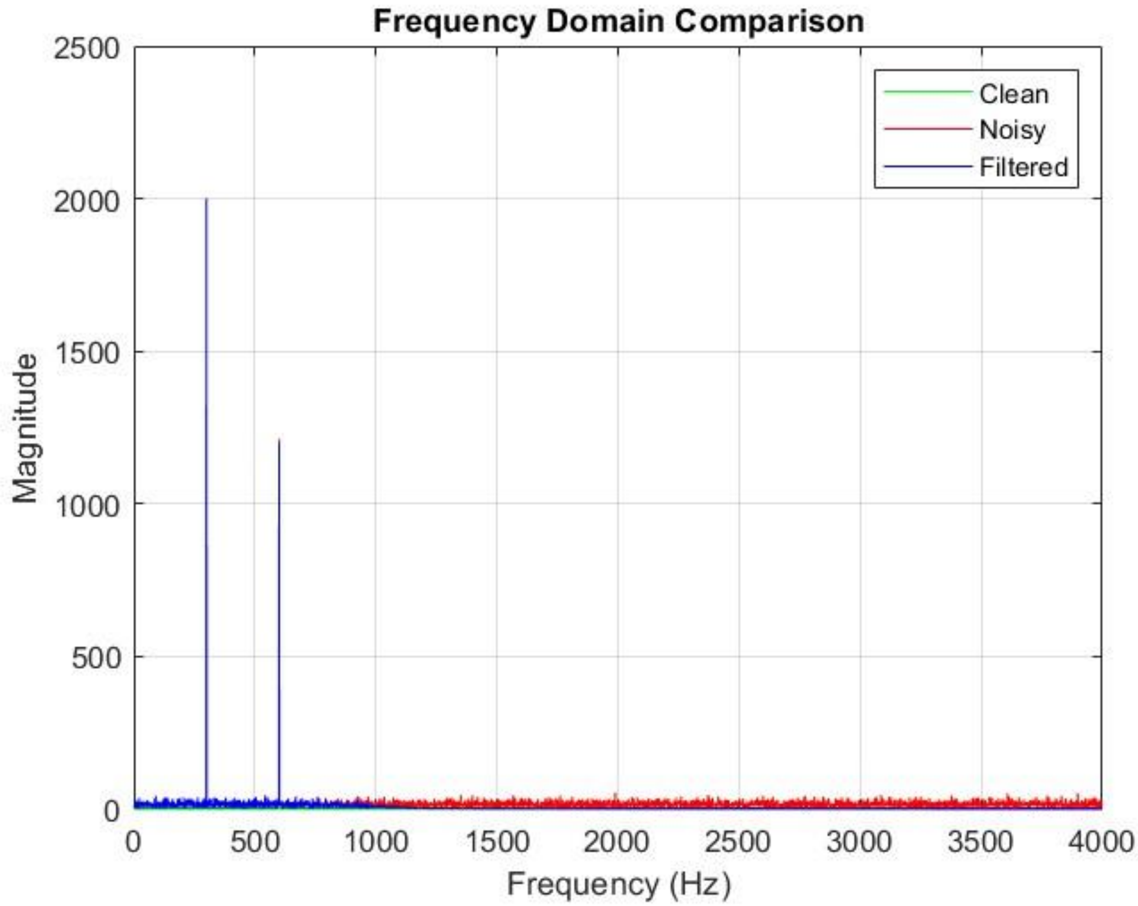

Figure 5: Comparison in the frequency domain.

Fast Fourier Transform (FFT): Used to calculate the frequency spectrum of signals. Display up to the Nyquist frequency: Only half of the spectrum (up to Fs/2) is shown, as the second half is symmetric. Frequency comparison plot: Clean, noisy, and filtered signals are plotted in a single figure. Colors green (clean), red (noisy), and blue (filtered) are used for distinction.

5. Conclusion

The project successfully demonstrated voice signal processing. The low-pass filter effectively reduced noise, and the filtered signal closely matched the clean signal. Time and frequency domain analyses provided insights into the effects of noise and filtering.

6. Future Work

- Use real audio files instead of synthetic signals

- Experiment with different filters (e.g., bandpass or FIR filters)

- Calculate quantitative metrics like Signal-to-Noise Ratio (SNR) or Mean Squared Error (MSE)

- Incorporate advanced analyses such as wavelet transforms or feature extraction