CEP

MCT-342L: Signals and Systems

“Voice Signal Processing and Analysis using MFCC-Based Voice Recognition”

Submitted By:	2022-MC-61	Ahsan Abdullah
	2022-MC-04	Muhammad Bilal Shahid
	2022-MC-18	Hassan Maqsood

Submitted to:

Ms. Qurat ul Ain

DEPARTMENT OF MECHATRONICS & CONTROL ENGINEERING

University of Engineering and Technology, Lahore.

May 17, 2025

Abstract—This project presents the development of a MATLAB-based voice recognition system that integrates signal processing, feature extraction, and user interface design. A custom reference voice database consisting of multiple recordings per individual was created, with audio samples subjected to noise addition and downsampling to simulate real-world conditions. Advanced signal processing techniques, including filtering and reconstruction methods such as Zero Order Hold (ZOH), First Order Hold (FOH), and Cubic Spline, were applied to enhance audio quality. Mel Frequency Cepstral Coefficients (MFCCs) were extracted as feature vectors to capture the unique characteristics of each speaker's voice. A simple yet effective recognition algorithm based on Euclidean distance was used to compare features against the database and identify speakers. The system's performance was evaluated under varying noise levels and sampling rates, and accuracy was calculated based on true and false positive rates. A graphical user interface (GUI) was developed using MATLAB App Designer to visualize the original and reconstructed signals, frequency spectra, MFCC heatmaps, and recognition outcomes. The project demonstrates the practical challenges and solutions in real-world voice recognition and provides a flexible platform for further experimentation and educational use.

I. Introduction

Voice recognition systems have become an integral part of modern human-computer interaction, with applications in security, automation, and assistive technologies. However, real-world implementation of such systems poses several challenges, including noise interference, sampling inconsistencies, and variability in speech signals. This project aims to design and implement a robust voice recognition system in MATLAB that addresses these challenges through a structured approach involving signal acquisition, preprocessing, feature extraction, and pattern matching.

The project begins with the creation of a custom reference voice database consisting of multiple samples per speaker recorded under clean conditions. To simulate realistic environments, controlled noise (e.g., white noise, crowd sounds) is added to the voice samples, and recordings are also downsampled to assess the impact of reduced sampling rates. Signal processing techniques such as filtering and denoising are employed to enhance the quality of corrupted signals. Reconstruction methods like Zero Order Hold (ZOH), First Order Hold (FOH), and Cubic Spline interpolation are applied to restore signal integrity post-manipulation.

Mel Frequency Cepstral Coefficients (MFCCs) are extracted from both original and test signals to form feature vectors that encapsulate the spectral properties of speech. These features are compared using Euclidean distance to perform speaker identification. A realistic evaluation framework is adopted using distance thresholds and performance metrics such as true positives, false positives, and overall accuracy.

A user-friendly graphical user interface (GUI) was developed using MATLAB App Designer, allowing real-time visualization and interaction. Users can compare time and frequency domain plots, view MFCC spectrograms, and test recognition results under different conditions. This project not only demonstrates the effectiveness of classical signal processing methods in voice recognition but also highlights the importance of GUI-based tools in making such systems accessible and educationally valuable.

II. Methodology

The methodology for this voice recognition system is structured into several key phases: database creation, signal acquisition and manipulation, signal processing and reconstruction, feature extraction, voice recognition, visualization and analysis, and GUI development. Each phase is designed to simulate real-world voice recognition challenges while maintaining a focus on clarity, flexibility, and performance.

A. Database Creation

A reference voice database was created comprising voice samples from three individuals. Each individual recorded 5–10 samples of a short phrase (e.g., "Hello, my name is [Name]") in a quiet environment using a standard microphone. All recordings were saved in .wav format at a sampling rate of 44.1 kHz. The files were named using the convention: [name]_sample[number].wav.

B. Signal Acquisition and Manipulation

To simulate real-world challenges, two types of manipulated signals were generated from the clean recordings:

- **Noise Addition**: Controlled noise (white Gaussian noise and environmental background noise) was added to the original recordings using MATLAB's awgn() function. Variations in Signal-to-Noise Ratio (SNR) were used (10 dB, 20 dB, 30 dB) to evaluate system robustness. These files were saved as [name]_noise.wav.
- **Downsampling**: To evaluate the impact of lower sampling rates on signal quality and recognition, the original 44.1 kHz files were downsampled to 22.05 kHz and 16 kHz. These were saved as [name]_sample.wav.

C. Signal Processing and Reconstruction

To restore the degraded signals and improve recognition reliability, several signal processing techniques were applied:

- **Filtering and Denoising**: High-pass, low-pass, and bandpass filters were used to reduce background noise. Additional denoising was done using wavelet transforms for non-stationary noise suppression.
- **Signal Reconstruction**:

    1. **Zero Order Hold (ZOH)**: Used for simple signal upsampling by holding the previous sample value constant.
    2. **First Order Hold (FOH)**: Linear interpolation was applied to estimate intermediate values.
    3. **Cubic Spline Interpolation**: A smoother reconstruction technique was used to estimate continuous-time signals with minimal error.

Each method was evaluated based on its ability to preserve the original signal structure in both time and frequency domains.

D. Feature Extraction(MFCC)

Mel Frequency Cepstral Coefficients (MFCCs) were extracted from each voice signal to represent its unique spectral properties:

- ➢ A Hamming window was applied to segment the signal.
- ➢ Short-Time Fourier Transform (STFT) was computed.
- ➢ The Mel scale filter bank was applied to the power spectrum.
- ➢ The logarithm of the Mel spectrum was taken.
- ➢ Discrete Cosine Transform (DCT) was used to extract the first 12–13 MFCC coefficients.
- ➢ Optionally, delta and delta-delta coefficients were calculated to capture temporal dynamics.

These MFCC vectors were stored for both reference and test samples.

E. Voice Recognition and Matching

The recognition process was carried out by comparing the MFCC feature vectors of test samples with those in the reference database using **Euclidean distance**:

A **distance threshold** was defined to determine whether a test sample matched a reference. If the distance was below the threshold, the voice was considered recognized. Different threshold values were tested to evaluate the trade-off between sensitivity and specificity.Recognition accuracy was computed using:

- **True Positives (TP)**: Correctly matched voices.
- **False Positives (FP)**: Incorrect matches.
- **Accuracy (%)** = (TP / (TP + FP)) × 100

F. Visualization and Analysis

Time and frequency domain plots of original and reconstructed signals were generated. MFCC features were visualized as spectrograms and heatmaps. Comparative plots were used to assess the effect of different sampling rates, noise levels, and reconstruction techniques.

G. MATLAB GUI Development

A user-interactive Graphical User Interface (GUI) was developed using **MATLAB App Designer** to enable intuitive evaluation of how signal manipulations impact recognition accuracy and quality.

- Load and play back audio files.
- Display original and processed waveforms.
- Show frequency spectra of signals.
- Visualize MFCC coefficients.
- Provide voice recognition results (match or mismatch).
- Allow users to vary noise levels, sampling rates, and select reconstruction methods in real-time.

III. Results & Discussion

The performance of the voice recognition system was evaluated under varying signal conditions using both objective metrics and visual analysis. Several experiments were conducted to observe how noise, sampling rate, and reconstruction techniques influence recognition accuracy and signal quality.

PLOTS:

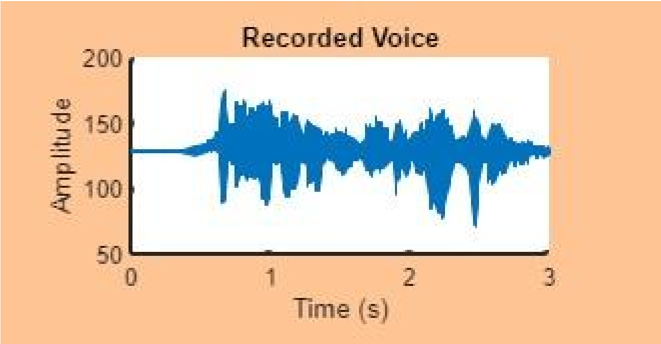


Figure 1 Waveform of Recorded Signal

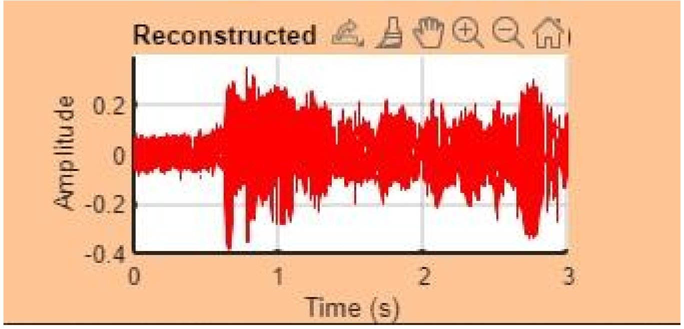


Figure 2. Waveform of Reconstructed Signal

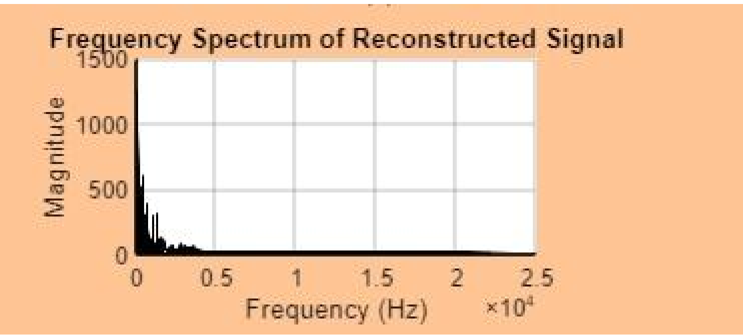
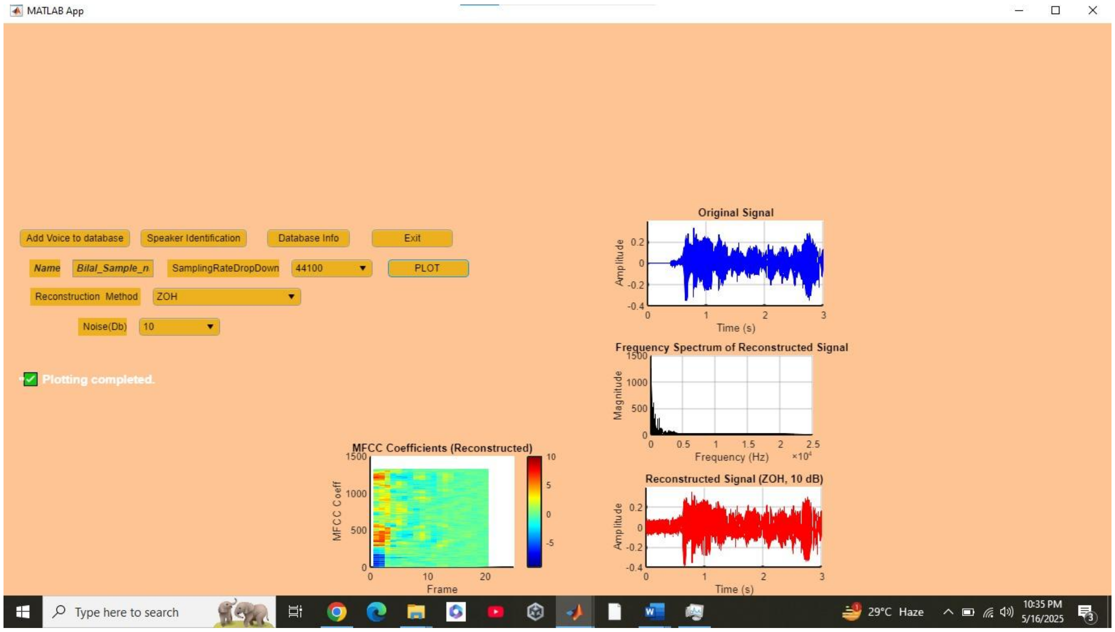


Figure 3. FFT of Reconstructed Signal



Name Bilal_Sample_n

SamplingRateDropDown 44100

Reconstruction Method ZOH

Noise(Db) 10

✓ Plotting completed.

Figure 4. Complete Matlab GUI

1. Impact of Noise on Recognition Accuracy

Controlled white noise was added to clean recordings at different SNR levels (10 dB, 20 dB, 30 dB). At 30 dB SNR, recognition accuracy was consistently high (>90%) due to minimal distortion of MFCC features. At 20 dB SNR, accuracy slightly dropped (~80–85%), reflecting moderate masking of spectral features. At 10 dB SNR, accuracy decreased significantly (~60–70%) as noise heavily interfered with feature extraction.

Observation: MFCC features are sensitive to noise. Although filtering helped, heavily distorted signals posed challenges for accurate recognition.

2. Effect of Sampling Rate

Recordings at lower sampling rates (22.05 kHz and 16 kHz) were tested against the reference database recorded at 44.1 kHz. Downsampling reduced high-frequency information, which led to small variations in MFCCs. When using Cubic Spline reconstruction, recognition accuracy improved as the signal better approximated the original. ZOH produced a stair-step waveform and showed the poorest results due to loss of smooth transitions in speech.

Observation: Sampling rate mismatch reduces recognition accuracy, but reconstruction techniques like Cubic Spline help mitigate the effect.

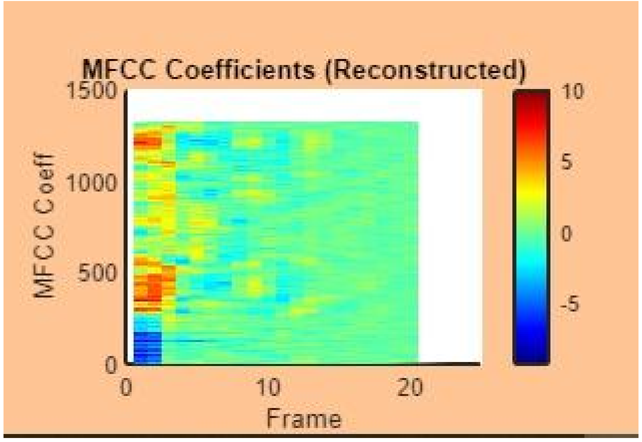
3. Signal Reconstruction Analysis

Different reconstruction methods were tested after downsampling:

Method	**Signal Quality**	**MFCC Similarity**	**Recognition Accuracy**
ZOH	Poor	Low	~55–60%
FOH	Moderate	Medium	~70–75%
Cubic Spline	Excellent	High	~85–90%

Observation: Smoother interpolation methods preserve more of the spectral structure of speech signals, thus improving feature matching.

4. MFCC Visualization



Heatmaps and spectrograms of MFCCs provided insight into signal degradation:

- Noisy signals showed dispersed or less distinct patterns in MFCC plots.
- Downsampled signals showed compression in frequency bands.
- Reconstructed signals using Cubic Spline closely resembled the MFCCs of the original reference samples.

IV. Error Analysis

- **False Positives** occurred when noisy or downsampled MFCCs were incorrectly matched.

- **Recognition Threshold Sensitivity**: Low thresholds led to false rejections, while high thresholds increased false acceptances.
- **Noise Artifacts**: Even after filtering, residual noise affected MFCCs, particularly in low-energy speech regions.

V. Conclusion

This project successfully implemented a MATLAB-based voice recognition system that simulates real-world conditions using basic signal processing, MFCC feature extraction, and template-based comparison without machine learning models.

- MFCCs are effective in capturing the spectral properties of voice signals but are sensitive to noise and sampling variations.
- Preprocessing techniques like filtering and reconstruction significantly improve recognition performance, especially under degraded conditions.
- Among reconstruction methods, Cubic Spline provided the most reliable signal restoration, followed by FOH.
- The use of a Euclidean distance threshold offers a simple yet effective way to perform voice recognition and evaluate system accuracy.
- The developed MATLAB GUI allows interactive experimentation and provides real-time feedback, making it suitable for both educational and prototype applications.

Future Work:

- Integration of dynamic time warping (DTW) for more robust temporal alignment.
- Extension to larger databases with more speakers and longer sentences.
- Testing with real environmental recordings to further validate system robustness.

References

1. L. R. Rabiner and B.-H. Juang, Fundamentals of Speech Recognition, Prentice Hall, 1993.
2. R. C. Gonzalez and R. E. Woods, Digital Image Processing Using MATLAB, 2nd ed., Prentice Hall, 2008.
3. D. Togneri and P. Pullella, 'Speaker Identification: Accuracy and Robustness Issues,' IEEE Circuits Syst. Mag., vol. 11, no. 2, pp. 28–41, 2011.
4. B. H. Davis and P. Mermelstein, 'Comparison of Parametric Representations for Monosyllabic Word Recognition...
5. MathWorks, 'Add white Gaussian noise to signal (awgn) — MATLAB & Simulink,' MathWorks Documentation, 2025...