nanda Sagar College of Engineering

Department of Electronics & Communication Engineering

(An Autonomous institution affiliated to VTU Belagavi, accredited by NBA & NAAC)
Shavige Malleshwara Hills, Kumaraswamy Layout, Bengaluru-560 111, Karnataka, India.

Program: B.E. **Branch:** ECE
Course: Digital Signal Processing **Semester and Section:** IV-D
Course code: 22EC43 **Date:**
Academic Year: 2023-2024 [EVEN] **Group No.:** 8

AAT Report

on

"Record a Speech Signal and do the Frequency analysis on the same."

.o.	USN	Name of the student	Simulation (4)	Report (3)	Presentation (3)	Marks (10)
1.	1DS22EC098	KAUSHIK K C				
2.	1DS22EC199	MANISH S				

Faculty Incharge:

Dr. Kumar P.
Assistant
professor,
ECE Dept.,
DSCE,
Bengaluru

Signature of the Faculty Incharge

CONTENT:

INTRODUCTION:

Speech signal is a type of sound signal that is generated during speech production. It includes various sounds, tones, and modulation. Speech signal are essential in various field such as telecommunications, voice recognition, and audio processing. Analyzing these signal provides the characteristics of the signal like pitch, tone, frequency analysis.

This simulation focuses on the recording of a speech signal and perform frequency analysis in the MATLAB. The MATLAB has effective tools that help to capture real-time audio, analyze its frequency components. The process of performing the frequency analysis involves the recording of the speech sample and applying the Fast Fourier Transform(FFT) to convert the time domain to its frequency- domain representation.

METHODOLOGY:

The process of recording a speech signal and performing the frequency analysis can be divided into two main parts:

- Recording the speech signal.
- Frequency analysis.

Recording the speech signal:

- Defining the sample rate and duration for the recording. The sample rate is used to determine the sample per second are captured, and the duration specifies how long the recording will last.
- Using MATLAB ‘ audiorecorder’ function to create an audio recorder object. The object is used to monitor the recording process with the defined sample rate and duration.
- The recordblocking function to start the recording. This function captures the recording and stores it in the recording object.

Frequency Analysis:

- Performing the fft to the audio data using the ‘fft’ function. This function converts the time- domain signal to the frequency domain representation.
- Calculate the magnitude of fft result to obtain the two-sided amplitude spectrum.

- Plot the frequency array that corresponds to single-sided amplitude spectrum.
- Then inorder to get back the same original speech signal the ifft (inverse fast fourier transform) function is used in the program.
- Then the reconstructed signal is plotted as shown.

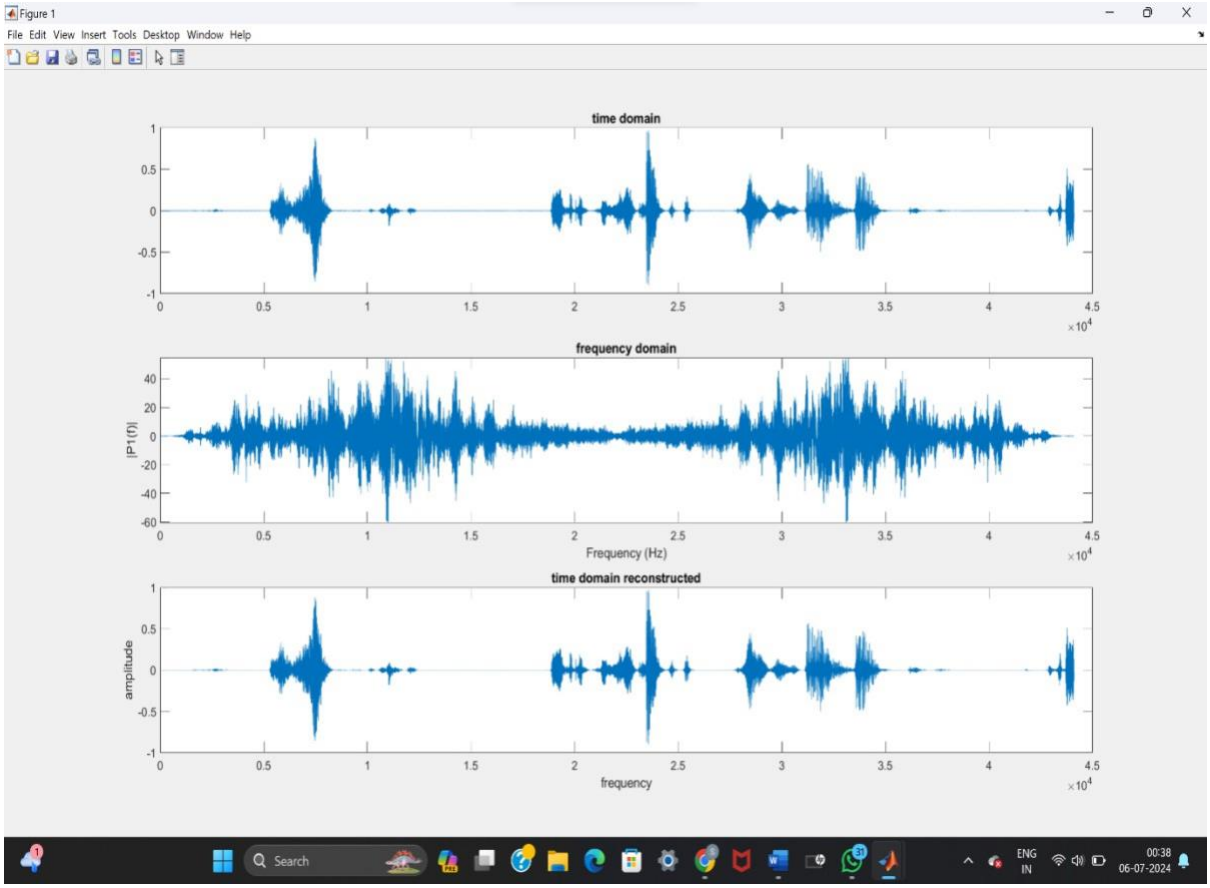
Through the above steps the recording of the speech signal and the frequency analysis can be performed effectively using MATLAB. This provides the spectral characteristics of the spectrum.

MATLAB CODE:

```
% Step 1: Define the sampling rate and the duration
fs = 4410; % Sampling rate in Hz
duration = 10; % Duration in seconds

% Step 2: Create an audiorecorder object
recObj = audiorecorder(fs, 16, 1);

% Step 3: Start recording
disp('Start speaking.');
recordblocking(recObj, duration);
disp('End of Recording.');

% playback the recording
play(recObj);

% Step 4: Get the audio data
audioData = getaudiodata(recObj);

% plot the waveform
subplot(3,1,1);
plot(audioData);
title('time domain');

% Step 5: Save the recorded audio to a file (optional)
audiowrite('speech_signal.wav', audioData, fs);

% Step 6: Load the audio file (if needed)
[audioData, fs] = audioread('speech_signal.wav');

% Step 7: Length of the audio data
L = length(audioData);

% Step 8: Perform FFT
Y = fft(audioData);

% Step 9: Compute the two-sided spectrum
P2 = abs(Y/L);

% Step 10: Compute the single-sided spectrum
P1 = P2(1:L/2+1);
P1(2:end-1) = 2*P1(2:end-1);

% Step 11: Define the frequency domain
```

```
f = fs*(0:(L/2))/L;

% Step 12: Plot the single-sided amplitude spectrum
subplot(3,1,2);
plot(real(Y));
title('frequency domain');
xlabel('Frequency (Hz)');
ylabel('|P1(f)|');

x= ifft(Y);
subplot(3,1,3);
plot(x);
xlabel('frequency');
ylabel('amplitude');
title('time domain reconstructed');
```

OUTPUT:



APPLICATIONS:

The recording and frequency analysis of the speech signal involves a wide range of applications across various fields. The key applications include,

1. Speech recognition.
2. Speaker identification and verification.
3. Audio compression.
4. Speech synthesis.
5. Speech therapy.
6. Communication system.