MATLAB Based Feature Extraction Using Mel Frequency Cepstrum Coefficients for Automatic Speech Recognition

Siddhant C. Joshi, Dr. A.N.Cheeran

Abstract— **Speech interface to computer is the next big step that the technology needs to take for general users. Automatic speech recognition (ASR) will play an important role in taking technology to the people. There are numerous applications of speech recognition such as direct voice input in aircraft, data entry, speech-to-text processing, voice user interfaces such as voice dialing. ASR system can be divided into two different parts, namely feature extraction and feature recognition.**
In this paper we present MATLAB based feature extraction using Mel Frequency Cepstrum Coefficients (MFCC) for ASR. MFCC algorithm makes use of Mel-frequency filter bank along with several other signal processing operations. Matrix of MFCC features obtained from our implementation of MFCC algorithm has number of rows equal to number of input frames and it is used in feature recognition stage.

Index Terms— **Automatic Speech Recognition, DFT, Feature Extraction, Mel frequency Cepstrum Coefficients, Spectral Analysis**

I. Introduction

Speech recognition is fundamentally a pattern recognition problem. Speech recognition involves extracting features from the input signal and classifying them to classes using pattern matching model. Performance of ASR system is measured on the basis of recognition accuracy, complexity and robustness. The deviation of operating conditions from those assumed during training phase may result in degradation of performance [1].
The feature extraction process aims to extract a compact, efficient set of parameters that represent the acoustic properties observed from input speech signal, for subsequent utilization by acoustic modeling. The feature extraction is a lossy (non-invertible) transformation. It is not possible to reconstruct the original speech from its features. [2].
There are three major types of feature extraction techniques, namely linear predictive coding (LPC), Mel frequency cepstrum coefficient (MFCC) and perceptual linear prediction (PLP). MFCC and PLP are the most commonly used feature extraction techniques in modern ASR systems [1].
This paper is organized as follows. Section II describes the feature extraction module. Section III describes spectral analysis using pre-emphasis, frame blocking and windowing. Section III discusses Mel frequency filter bank, which is at the heart of the MFCC algorithm. Finally, section V concludes the paper.

II. Feature Extraction Module

At the core of the feature extraction lies the short-term spectral analysis (e.g. discrete Fourier transform), accompanied with several signal processing operations. The basic principle here is to extract a sequence of features for each short-time frame of the input signal, with an assumption that such a small segment of speech is sufficiently stationary to allow meaningful modeling [3]. The efficiency of this phase is important for the next phase since it affects the behavior of modeling process. We use the Mel-frequency Cepstral Coefficients (MFCC) for feature extraction.
The speech waveform, sampled at 8 kHz is used as an input to the feature extraction module. Software 'Audacity' is used to record the input speech database. In MATLAB, 'wavread' function reads the input wave file and returns its samples. Speech files are recorded in 'wave' format, with the following specifications: Fs = Sample rate in Hertz = 8000 and n = Number of bits per sample = 16. Figure 1 shows block diagram of the feature extraction processing.

III. Spectral Analysis

Spectral analysis is concerned with determining the frequency content of an arbitrary signal. Feature extraction is done on short time basis. The speech signal is divided into overlapped fixed length frames. A set of cepstrum domain or frequency domain parameters, called feature vector are derived from each frame. Different signal processing operations such as pre-emphasis, framing, windowing and Mel cepstrum analysis are performed on the input signal, at different stages of the MFCC algorithm [4].

A. Pre-emphasis

Noise has a greater effect on the higher modulating frequencies than the lower ones. Hence, higher frequencies are artificially boosted to increase the signal-to-noise ratio. Pre-emphasis process performs spectral flattening using a first order finite impulse response (FIR) filter [1], [3]. Equation (1) represents first order FIR filter.

$$H(z) = 1 - \alpha z^{-1}, 0.9 \le \alpha \le 1.0 \quad (1)$$

B. Frame blocking

Speech is a non-stationary signal. If the frame is too long, signal properties may change too much across the window, affecting the time resolution adversely. If the frame is too short, resolution of narrow-band components will be sacrificed, affecting the frequency resolution adversely.

ISSN: 2278 – 7798

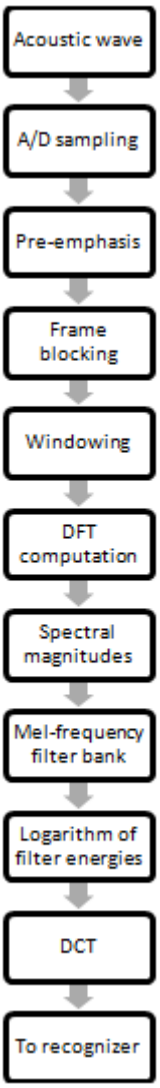

Figure 1: Block diagram of the feature extraction processing

There is a trade-off between time resolution and frequency resolution [1], [3]. We choose number of samples in each frame as 256, with the number of samples overlapping between adjacent frames as 128. Overlapping frames are used to capture information that may occur at the frame boundaries. Number of frames is obtained by dividing the total number of samples in the input speech file by 128.
For covering all samples of input, last frame may require zero padding. All frames are stored as rows in one single matrix with number of rows equal to number of frames and number of columns equal to 256, which is also equal to the frame width.

C. *Windowing*

Discontinuities at the beginning and end of the frame are likely to introduce undesirable effects in the frequency response. Hence, each row is multiplied by window function. A window alters the signal, tapering it to nearly zero at the beginning and the end [1], [3].
We use Hamming window as, it introduces the least amount of distortion. Our implementation uses Hamming window of length 256. Equation (2) shows the discrete time domain representation of Hamming window function.

$$h[\mathrm{n}] = \begin{cases} 0.54 - 0.46\cos(2\pi n / N), 0 \le n \le N \\ 0, otherwise \end{cases} \quad (2)$$

D. *Spectral magnitude of DFT*

Spectral information means the energy levels at different frequencies in the given window. Time domain data is converted into frequency domain to obtain the spectral information. Time domain data is converted to frequency domain by applying Discrete Fourier Transform (DFT) on it [5]. Equation (3) represents DFT.

$$X(\mathrm{k}) = \sum_{n=0}^{N-1} x(\mathrm{n}) e^{-j2\pi kn/N}, 0 \le k \le N-1 \quad (3)$$

Here, x(n) represents input frame of 256 samples and X(k) represents its equivalent DFT. We use 256-point FFT algorithm to convert each frame of 256 samples into its equivalent DFT.
FFT output is a set of complex numbers i.e. real and imaginary parts. Speech recognition systems deal with real data. Hence, complex value is always ignored [1]. If we assume the real and imaginary parts of X(k) as Re(X(k)) and Im(X(k)), then the spectral magnitude of the speech signal can be obtained by using equation (4). Spectral magnitudes of each frame are stored as rows in one single matrix with number of rows equal to number of frames and number of columns equal to 256, which is also equal to the frame width.

$$|X(\mathrm{k})| = \sqrt{(\mathrm{Re}(\mathrm{X}(\mathrm{k})))^2 + (\mathrm{Im}(\mathrm{X}(\mathrm{k})))^2} \quad (4)$$

IV. Mel Frequency Filter Bank

Mel-frequency analysis of speech is based on human perception experiments. It has been proved that human ears are more sensitive and have higher resolution to low frequency compared to high frequency. Hence, the filter bank is designed to emphasize the low frequency over the high frequency [1], [3].
Also the voice signal does not follow the linear frequency scale used in FFT. Hence, a perceptual scale of pitches equal in distance, namely Mel scale is used for feature extraction. Mel scale frequency is proportional to the logarithm of the linear frequency, reflecting the human perception [1]. We use log because our ears work in decibels. Figure 2 shows frequencies in Mel scale plotted against frequencies in linear scale. Equation (5) is used to convert linear scale frequency into Mel scale frequency.

$$Mel(\mathrm{f}) = 2595\log_{10}\left(1 + \frac{f}{700}\right) \quad (5)$$

Triangular band pass filters are used to extract the spectral envelope, which is constituted by dominant frequency components in the speech signal. Thus, Mel-frequency filters are triangular band pass filters non-uniformly spaced on the linear frequency axis and uniformly spaced on the Mel frequency axis, with more number of filters in the low frequency region and less number of filters in the high frequency region [1], [6].

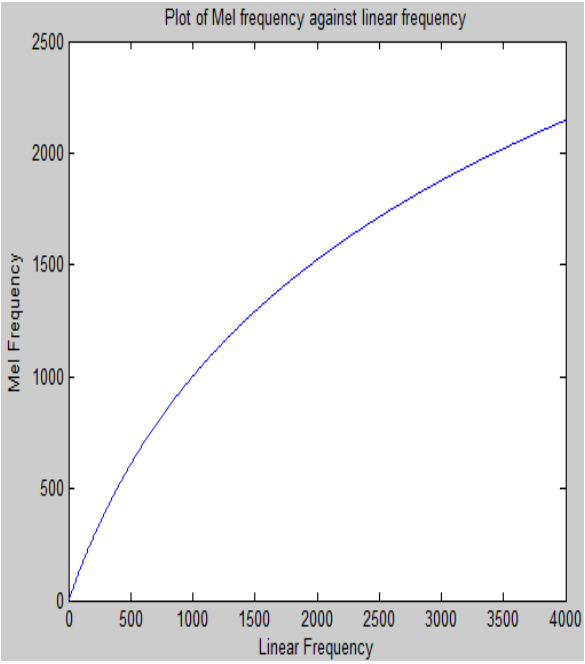

Figure 2: Plot of Mel frequencies against linear frequencies

Magnitude response of each filter is equal to unity at the center and decreases linearly to zero at the center frequencies of two adjacent filters. We use Mel frequency filter bank with 20 triangular overlapping filters. Maximum modulating frequency is 4000 Hz. Hence the maximum frequency in Mel scale is 2146.1. Also, the bandwidth of each triangular band pass filter in Mel scale is 204.39. Centers of triangular band pass filters in Mel frequency domain and corresponding centers in linear frequency domain are shown in table I.

Table I: Centers of triangular band pass filters in Mel frequency domain and linear frequency domain

Serial Number	Center of BPF in Mel frequency domain	Center of BPF in linear frequency domain
1	.102.2	66.4
2	204.4	139.2
3	306.6	218.8
4	408.8	306.1
5	511	401.5
6	613.2	506.1
7	715.4	620.6
8	817.5	745.9
9	919.7	883.2
10	1021.9	1033.4
11	1124.1	1198
12	1226.3	1378.1
13	1328.5	1575.4
14	1430.7	1791.3
15	1532.9	2027.8
16	1635.1	2286.7
17	1737.3	2570.2
18	1839.5	2880.6
19	1941.7	3220.5
20	2043.9	3592.6

Table II: Center frequencies, lower cut-off frequencies and upper cut-off frequencies for triangular band pass filters

Serial Number	Center frequency	Lower cut-off frequency	Upper cut-off frequency
1	4	1	8
2	8	4	14
3	14	8	19
4	19	14	25
5	25	19	32
6	32	25	39
7	39	32	47
8	47	39	56
9	56	47	66
10	66	56	76
11	76	66	88
12	88	76	100
13	100	88	114
14	114	100	129
15	129	114	146
16	146	129	164
17	164	146	184
18	184	164	206
19	206	184	229
20	229	206	256

We do not have the frequency resolution required to put the filters at the exact linear frequency points, since we are processing the data in discrete frequency domain. Hence, we round these linear frequencies to nearest FFT points.

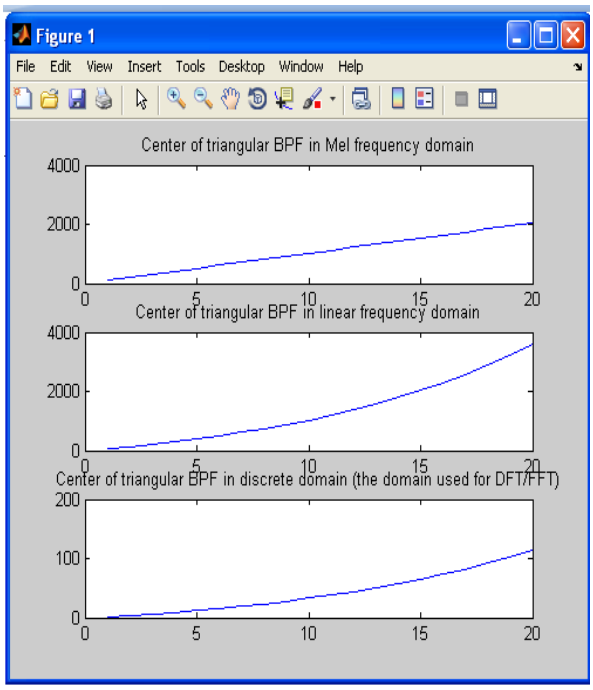

Figure 3: Plot centers of triangular band pass filters in Mel frequency domain, linear frequency domain and discrete frequency domain

ISSN: 2278 – 7798

Center frequencies, lower cut off frequencies and upper cut off frequencies for the triangular band pass filters are shown in table II. Figure 3 shows center of triangular band pass filters in Mel frequency domain, linear frequency domain and discrete frequency domain plotted against m i.e. number of triangular band pass filter in the filter bank. It is observed that as m increases, the difference between centers of two adjacent filters increases in linear scale and remains the same in Mel scale.

Equation (6) represents the filter bank with M (m = 1, 2, 3….M) filters, where m is the number of triangular filter in the filter bank [1].

$$Hm(\mathrm{k}) = \begin{cases} 0, \text{for } k < f(\mathrm{m}-1) \\ \dfrac{k - f(\mathrm{m}-1)}{\mathrm{f(m)} - \mathrm{f(m}-1)}, \text{for } f(\mathrm{m}-1) \le \mathrm{k} \le f(\mathrm{m}) \\ \dfrac{f(\mathrm{m}+1) - \mathrm{k}}{f(\mathrm{m}+1) - \mathrm{f(m)}}, \text{for } f(\mathrm{m}) \le \mathrm{k} \le \mathrm{f(m}+1) \\ 0, \text{for } k > f(\mathrm{m}+1) \end{cases} \quad (6)$$

Each triangular filter in the filter bank satisfies equation (7).

$$\sum_{m=0}^{M-1} Hm(\mathrm{k}) = 1 \quad (7)$$

We use 20 filters. Hence, M = 20. FFT spectrum is passed through Mel filters to obtain Mel spectrum. Mel frequency filter bank is applied to every 256 samples frame and the filtered response is computed. In frequency domain, filtering is obtained by multiplying the FFT of signal and transfer function of the filter on element by element basis.

A. Logarithm of filter energies

Human ears smooth the spectrum and use the logarithmic scale approximately. We use equation (8) to compute the log-energy i.e. logarithm of the sum of filtered components for each filter [1], [3].

$$S(\mathrm{m}) = \log_{10}\left[\sum_{k=0}^{N-1} |X(\mathrm{k})|^2 .Hm(\mathrm{k})\right], 0 \le m \le M \quad (8)$$

Thus, each bin per frame per filter holds the log-energy obtained by computing logarithm of weighted sum of spectral magnitudes in that filter-bank channel. Hence, we get 20 numeric values for each frame at the output of this stage. Output of this stage is stored in a matrix with number of rows equal to number of frames and number of columns equal to 20 i.e. number of filters in the filter bank.

B. Discrete cosine transform

The discrete cosine transform (DCT) converts the log power spectrum (Mel frequency domain) into time domain [7]. DCT gathers most of the information of the signal to its lower order coefficients, resulting in significant reduction in computational cost [1]. Equation (9) represents the discrete cosine transform.

$$C(\mathrm{k}) = \sum_{m=0}^{M-1} S(\mathrm{m}) \cos(\pi\, \mathrm{k(m}+1/2)/\mathrm{M}), 0 \le \mathrm{k} < \mathrm{K} \quad (9)$$

Here, value of K ranges between 8 and 13. We choose K as 13. Hence, we obtain 13 coefficients for each frame. At the output of this stage, we get a matrix with number of rows equal to the number of frames and number of columns equal to K = 13. Thus, cepstral analysis is performed on Mel-spectrum to obtain Mel Frequency Cepstrum Coefficients (MFCC).

V. Conclusion

Mel frequency filter bank of 20 non-linearly spaced, triangular band pass filters with overlapping bandwidths; when applied to the spectral magnitudes of FFT yields the dominant frequency components (or peaks or formants) for each frame of the input speech signal. MFCC algorithm performed on a 'wave' file in MATLAB yields a matrix with number of rows equal to number of frames, which is determined by the size of input file and number of columns equal to the DCT size, which is 13 in our case.

REFERENCES

[1] Yuan Meng, *Speech recognition on DSP: Algorithm optimization and performance analysis,* The Chinese university of Hong Kong, July 2004, pp. 1-18.

[2] Aldebaro Klautau, *The MFCC*, 11/12/05, pp. 1-5.

[3] Lindasalwa Muda, Mumtaj Begam and I. Elamvazuthi, *Voice recognition algorithm using MFCC & DTW techniques*, Journal Of Computing, Volume 2, Issue 3, March 2010, ISSN 2151-9617, pp. 138-143.

[4] Vibha Tiwari, *MFCC and its applications in speaker recognition*, International Journal on Emerging Technologies 1(1): (2010), pp. 19-22.

[5] Ripul Gupta, *Speech recognition for Hindi*, Indian Institute of Technology, Bombay, pp. 11-14.

[6] Sirko Molau, Michael Pitz, Ralf Schl¨uter, and Hermann Ney, *Computing Mel-frequency cepstral coefficients on the power spectrum*, University of Technology, 52056 Aachen, Germany

[7] Gaurav, Devanesamoni Shakina Deiv, Gopal Krishna Sharma, Mahua Bhattacharya, *Development of Application Specific Continuous Speech Recognition System in Hindi*, Journal of Signal and Information Processing, 2012, 3, pp. 394-401.