International Research Journal of Engineering and Technology (IRJET) e-ISSN: 2395-0056
Volume: 02 Issue: 06 | Sep-2015 www.irjet.net p-ISSN: 2395-0072

GENERATING AN ISOLATED WORD RECOGNITION SYSTEM USING MATLAB

Pinaki Satpathy[1*], Avisankar Roy[1], Kushal Roy[1], Raj Kumar Maity[1], Surajit Mukherjee[1]

[1] Asst. Prof., Electronics and Communication Engineering, Haldia Institute Of Technology, West Bengal, India
[*]Corresponding Author: pinakihit.sat@gmail.com

Abstract - MATLAB's straight forward programming interface makes it an ideal tool for speech analysis. In this work, experience was gained in general MATLAB programming. A basic speaker recognition algorithm has been written to sort through a rule base in MATLAB and choose the one most likely match based on the pre define time frame of the speech utterance. Speech communication has evolved to be efficient and robust and it is clear that the route to computer based speech recognition is the modeling of the human system.

Speaker dependent speech recognition is therefore an engineering compromise between the ideal, i.e. a complete model of the human, and the practical, i.e. the tools that science and technology provide and that costs allow the modeling of the human system.

Key Words: *Mel frequency cepstrum, Mel frequency wrapping, Mat lab*

1. INTRODUCTION

Speaker recognition is the process of automatically recognizing who is speaking on the basis of individual information included in speech waves. ***Speech recognition*** (SR) is the translation of spoken words into text. It is also known as "automatic speech recognition" (ASR), "computer speech recognition", or just "speech to text" (STT). These systems analyze the person's specific voice and use it to fine-tune the recognition of that person's speech, resulting in more accurate transcription [1-6]. Systems that do not use training are called "speaker-independent" systems. Systems that use training are called "speaker-dependent" systems. Our aim in this project is to design an ***"Isolated Word Reorganization System Using Mat lab"*.**

1.1 Mel-frequency cepstrum coefficients processor:

A block diagram of the structure of an MFCC processor is given in Figure 3. The speech input is typically recorded at a sampling rate above 10000 Hz. This sampling frequency was chosen to minimize the effects of *aliasing* in the analog-to-digital conversion. These sampled signals can capture all frequencies up to 5 kHz, which cover most energy of sounds that are generated by humans. As been discussed previously, the main purpose of the MFCC processor is to mimic the behaviour of the human ears. In addition, rather than the speech waveforms themselves, MFFC‟s are shown to be less susceptible to mentioned variations.

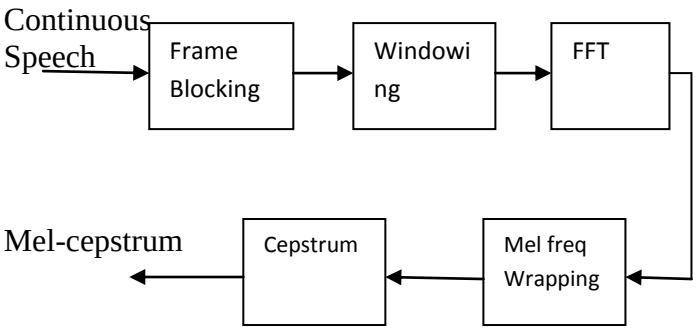

Figure 1 Block diagram of the MFCC processor.

1.2 Mel-frequency wrapping:

As mentioned above, psychophysical studies have shown that human perception of the frequency contents of sounds for speech signals does not follow a linear scale. Thus for each tone with an actual frequency, *f*, measured in Hz, a subjective pitch is measured on a scale called the "Mel" scale. The *Mel-frequency* scale is linear frequency spacing below 1000 Hz and a logarithmic spacing above 1000 Hz. One approach to simulating the subjective spectrum is to use a filter bank, spaced uniformly on the mel-scale (see Figure 2).That filter bank has a triangular bandpass frequency response, and the spacing as well as the bandwidth is determined by a constant mel frequency interval. The number of mel spectrum coefficients, *K*, is typically chosen as 20. Note that this filter bank is applied in the frequency domain, thus it simply amounts to applying the triangle-shape windows as in the Figure 2 to the spectrum. A useful way of thinking about this mel-wrapping filter bank is to view each filter as a histogram bin (where bins have overlap) in the frequency domain.

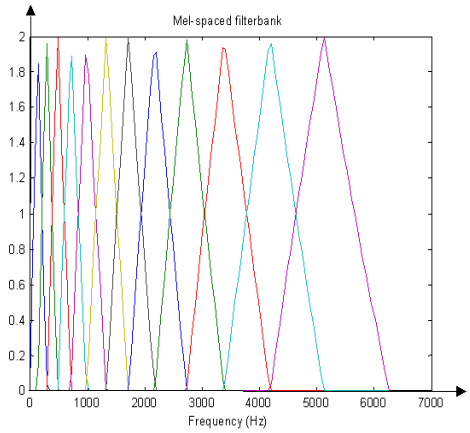

Figure 2 An example of mel-spaced filterbank

One approach to simulating the subjective spectrum is to use a filter bank, spaced uniformly on the mel-scale (see Figure 2).That filter bank has a triangular bandpass frequency response, and the spacing as well as the bandwidth is determined by a constant mel frequency interval. The number of mel spectrum coefficients, *K*, is typically chosen as 20. Note that this filter bank is applied in the frequency domain, thus it simply amounts to applying the triangle-shape windows as in the Figure 2 to the spectrum. A useful way of thinking about this mel-wrapping filter bank is to view each filter as a histogram bin (where bins have overlap) in the frequency domain.

2. WAVEFORM COMPARISON

Using the results and information learned from pitch and formant analysis, a waveform comparison code was written. Speech waveform files can be such criteria that can be used to characterize a speech file. The slow speech file was used as a reference file. Four sorting routines were then written to compare the files. The sorting routines performed the following functions: sort and compare the average pitch of the reference file, compare the formant vector of the reference file to all wav files, sort for the top 20 average pitch correlations and then sort these files by formant vectors, and finally to sort for the top 20 formant vector correlations and then sort these by average pitch. Sample code for the case of comparing the average pitch and then comparing the top 12 most likely matches by formant peak difference vectors. The three other cases use code characterized based on various criteria. Average pitch and formant peak position vectors are two from this sample to achieve their results.Figure.3 shows the wave comparison result. The upper plot shows the file with natural background noise. The noise signal is more prevalent in the middle figure which shows the shifted FFT of the original signal.

Noise can be seen as a broad peak at approximately $1x10^4$ Hz, as well as an overall background component. The bottom figure shows the signal after application of a 3rd order Butterworth filter and amplitude scaling to yield a valid comparison to the original signal.

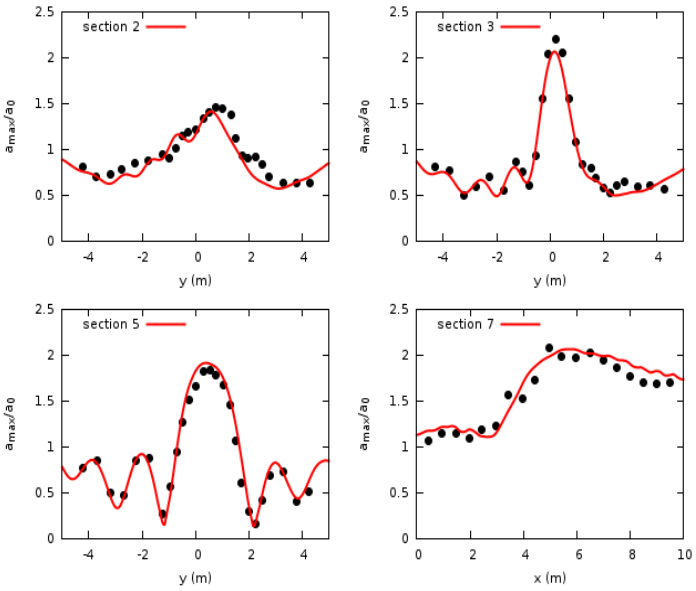

Figure 3: Plot for wave comparison result.

3. RESULT

Speech-recognition technology is embedded in voice-activated routing systems at customer call centres, voice dialling on mobile phones, and many other everyday applications. A robust speech-recognition system combines accuracy of identification with the ability to filter out noise and adapt to other acoustic conditions, such as the speaker's speech rate and accent. Designing a robust speech-recognition algorithm is a complex task requiring detailed knowledge of signal processing and statistical modelling.

This article demonstrates a workflow that uses built-in functionality in MATLAB® and related products to develop the algorithm for an isolated digit recognition system. The system is speaker-dependent—that is, it recognizes speech only from one particular speaker's voice.

Developing the Acoustic Model

A good acoustic model should be derived from speech characteristics that will enable the system to distinguish between the different words in the dictionary.

We know that different sounds are produced by varying the shape of the human vocal tract and that these different sounds can have different frequencies. To investigate these frequency characteristics we examine the power spectral density (PSD) estimates of various spoken digits. Since the human vocal tract can be modelled as an all-pole filter, we use the Yule-Walker parametric spectral estimation technique from Signal Processing Toolbox™ to calculate these PSDs.

After importing an utterance of a single digit into the variable 'speech', we use the following MATLAB code to visualize the PSD estimate:

```
order = 12;
nfft = 512;
Fs = 8000;
pyulear(speech,order,nfft,Fs);
```

Since the Yule-Walker algorithm fits an autoregressive linear prediction filter model to the signal, we must specify an order of this filter. We select an arbitrary value of 12, which is typical in speech applications.

Figures 4a and 4b plot the PSD estimate of three different utterances of the words 'one' and 'two'. We can see that the peaks in the PSD remain consistent for a particular digit but differ between digits. This means that we can derive the acoustic models in our system from spectral features.

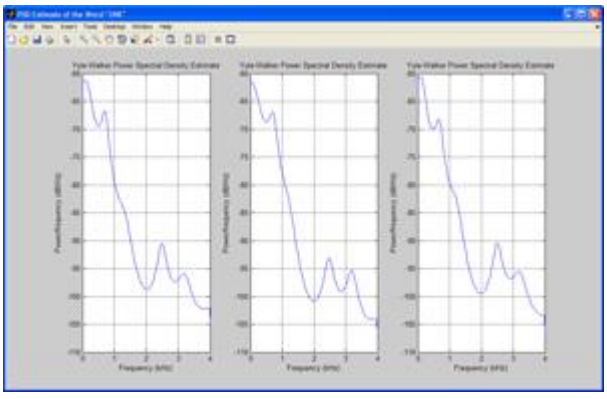

Figure 4a. Yule Walker PSD estimate of three different utterances of the word "ONE."

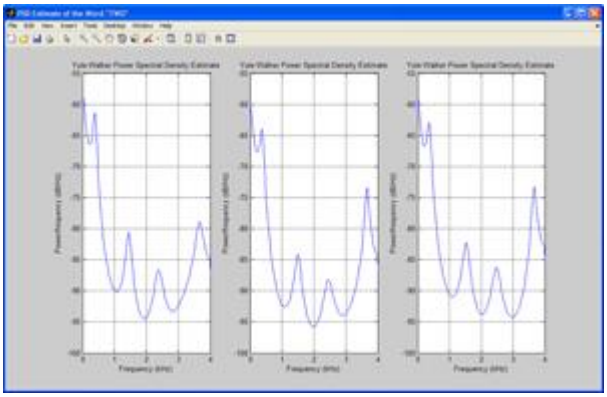

Figure 4b. Yule Walker PSD estimate of three different utterances of the word "TWO."

From the linear predictive filter coefficients, we can obtain several feature vectors using Signal Processing Toolbox functions, including reflection coefficients, log area ratio parameters, and line spectral frequencies.

One set of spectral features commonly used in speech applications because of its robustness is Mel Frequency Cepstral Coefficients (MFCCs). MFCCs give a measure of the energy within overlapping frequency bins of a spectrum with a warped (Mel) frequency scale[1].

Since speech can be considered to be short-term stationary, MFCC feature vectors are calculated for each frame of detected speech. Using many utterances of a digit and combining all the feature vectors, we can estimate a multidimensional probability density function (PDF) of the vectors for a specific digit. Repeating this process for each digit, we obtain the acoustic model for each digit.

During the testing stage, we extract the MFCC vectors from the test speech and use a probabilistic measure to determine the source digit with maximum likelihood. The challenge then becomes to select an appropriate PDF to represent the MFCC feature vector distributions.

Figure 5a shows the distribution of the first dimension of MFCC feature vectors extracted from the training data for the digit 'one'.

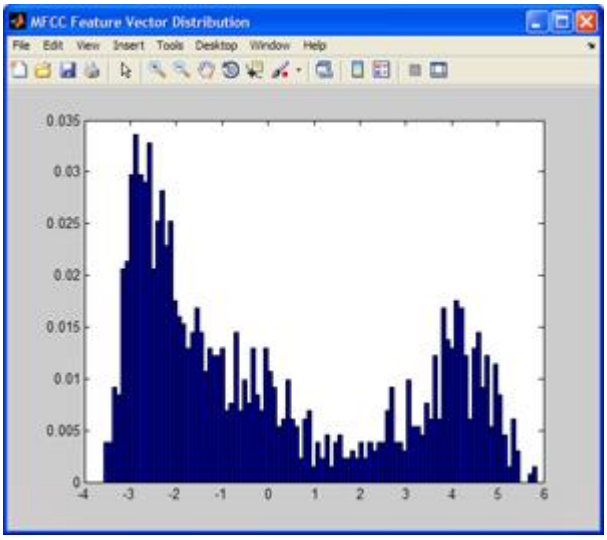

Figure 5a. Distribution of the first dimension of MFCC features vectors for the digit 'one.'

One solution is to fit a Gaussian mixture model (GMM), a sum of weighted Gaussians (Figure 5b).

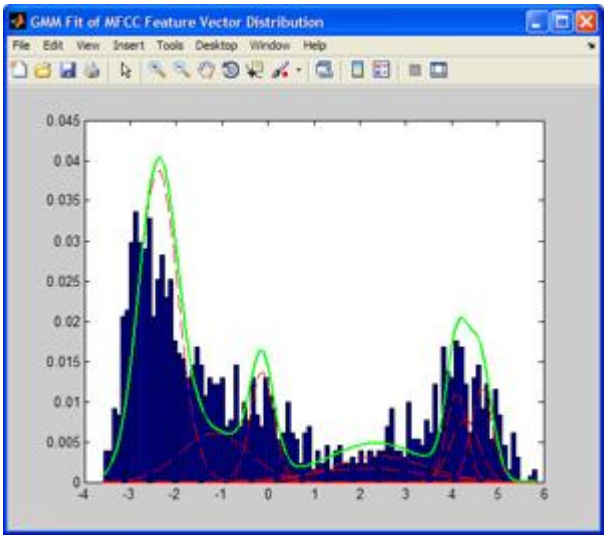

Figure5b. Overlay of estimated Gaussian components (red) and overall Gaussian mixture model (green)

The complete Gaussian mixture density is parameterized by the mixture weights, mean vectors, and covariance matrices from all component densities. For isolated digit recognition, each digit is represented by the parameters of its GMM.

To estimate the parameters of a GMM for a set of MFCC feature vectors extracted from training speech, we use an iterative expectation-maximization (EM) algorithm to obtain a maximum likelihood (ML) estimate. Given some MFCC training data in the variable MFCCtraindata, we use the Statistics and Machine Learning Toolbox gmdistribution function to estimate the GMM parameters. This function is all that is required to perform the iterative EM calculations.

```
%Number of Gaussian component densities
M = 8;
model = gmdistribution.fit (MFCCtraindata,M);
```

Building the User Interface

After developing the isolated digit recognition system in an offline environment with prerecorded speech, we migrate the system to operate on streaming speech from a microphone input. We use MATLAB GUIDE tools to create an interface that displays the time domain plot of each detected word as well as the classified digit (Figure 6).

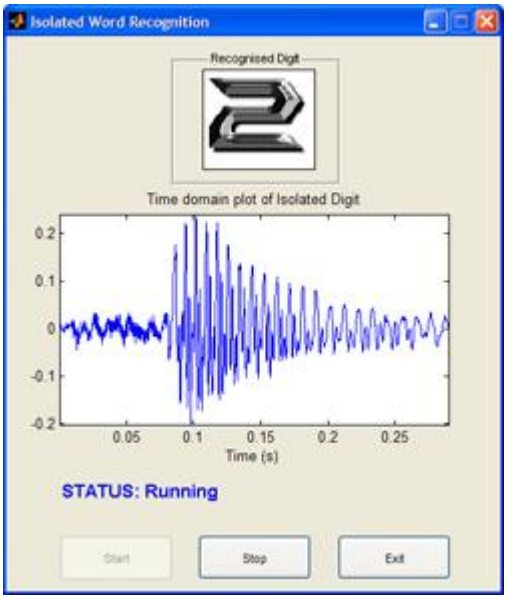

Figure 6. Interface to final application.

Extending the Application

The algorithm described in this article can be extended to recognize isolated words instead of digits, or to recognize words from several speakers by developing a speaker-independent system.

If the goal is to implement the speech recognition algorithm in hardware, we could use MATLAB and related products to simulate fixed-point effects, automatically generate embedded C code, and verify the generated co

4. CONCLUSIONS

A crude speaker recognition code has been written using the MATLAB programming language. This code uses comparisons between the average pitch of a recorded wav file in the PSD of each file. It was found that comparison based on pitch produced the most accuracy but could likely be improved. Experience was also gained in speech editing as well as basic filtering techniques. While the methods utilized in the design of the code for this project are a good foundation for a speaker recognition system, more advanced techniques would have to be used to produce a successful speaker recognition system. Speaker recognition involves the speaker identification to output the identity of the person most likely to have spoken from among a given population or to verify a person's identity who he/she claims to be from a given speech input. Since this recognition system is used for security, then an ethical consideration would involve making sure the system is up to standard so that imposters cannot be accepted.

We concluded that in this project speech of a specified speaker is recognised and verified successfully using all basic principles of speech analysis and speaker recognition method and become aware of its wide applications and benefits on mankind.

ACKNOWLEDGEMENT

The work is carried out through the research facility at the Department of Electronics and Communication Engineering, Haldia Institute Of Technology, Haldia, West Bengal. The Authors also would like to thank the authorities of HIT, Haldia for encouraging this research work. Our thanks to the experts who have contributed towards development of this paper.

REFERENCES

[1] Z. Ghahramani and M.I. Jordan, "Factorial Hidden Markov Models," *Machine Learning*, 29, pp. 245-275, 1997.

[2] B. Logan and P. Moreno, "Factorial HMMs for Acoustic Modeling," *ICASSP*, pp. 813-816, 1998.

[3] S. Zhang, C. Zhu, J. K. O. Sin, and P. K. T. Mok, "A novel ultrathin elevated channel low-temperature poly-Si TFT," *IEEE Electron Device Lett.*, vol. 20, pp. 569–571, Nov. 1999.

[4] S.B. Davis and P. Mermelstein, "Comparison of Parametric Representations for Monosyllabic Word Recognition in Continuously Spoken Sentences," IEEE Transactions on Acoustics, Speech and Signal Processing, Vol. ASSP-28, No. 4, August 1980.

[5] L.R. Rabiner, "A Tutorial on Hidden Markov Models and Selected Applications in Speech Recognition," *Proceedings of the IEEE*, Vol. 77, No. 2, February 1989.

[6] H.K. Kim and R.C. Rose, "Cepstrum-Domain Acoustic Freature Compensation Based on Decomposition of Speech and Noise for ASR in Noisy Environments," *IEEE Transactions on Speech and Audio Processing*, Vol. 11, No. 5, September 2003