International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 4 Issue 07, July-2015

Comparative Study of Isolated Word Recognition System for Hindi Language

Suman K. Saksamudre
Dept of Computer Science & IT
Dr. B. A. M. University
Aurangabad 431004, India

R. R. Deshmukh
Dept of Computer Science & IT
Dr. B. A. M. University
Aurangabad 431004, India

***Abstract*— Speech is a natural way of communication and it provides an easy user interface to machines so that automatic speech recognition (ASR) system is considered as necessary. But the automatic speech recognition (ASR) system doesn't perform perfectly for any language. So that generation of an accurate and robust acoustic model is necessary. The overall performance of any speech recognition system is highly depends on the feature extraction technique and classifier. In this paper, we have described Comparative study Isolated Word Recognition System for Hindi Language using MFCC as feature extraction and KNN as pattern classification technique. When our system is trained for First 10 words it achieves 89% rate of recognition and when trained for all 100 words it achieves 62.50% rate of recognition. As vocabulary increases performance decreases.**

***Keywords*— Feature extraction, MFCC, DCT, FFT, vocabulary, KNN .**

I. INTRODUCTION

Automatic recognition of speech by machine has been a goal of research for more than four decades. In the world of science, computer has always understood human mimics. The idea which generated for making speech recognition system is because it is convenient for humans to interact with a computer, robot or any machine through speech or vocalization rather than difficult instructions [1]. Human beings have long been inspired to create computer that can understand and talk like human. Since, 1960s computer scientists have been researching various ways and means to make computer record, interpret and understand human speech [2].

Speech recognition is the process by which computer maps an acoustic speech signal to some form of abstract meaning of the speech. This process is highly difficult [3] since sound has to be matched with stored sound bites on which further analysis has to be done because sound bites do not match with pre-existing sound pieces. Various feature extraction methods and pattern matching techniques are used to make better quality speech recognition systems. Feature extraction technique and pattern matching techniques plays important role in speech recognition system to maximize the rate of speech recognition of various persons.

There are two main phases in a speech recognition system [4]: training phase and testing phase for recognition. During the training phase, first off all features are extracted from the all speech signals using various feature extraction techniques such as MFCC, LPC, LDA and RASTA [5] etc. These features are in the form of vector. In this way a training vector is generated from the speech signal of each word spoken by the user. The training vector has the spectral features which distinguishes different words based on its class. These extracted features are the main component of whole speech recognition system. Each training vector can serve as a template for a single word or a word class. This training vector is used in the next phase of recognition. During the recognition phase, the user speaks any word for which the system was trained. A test pattern is generated for that word. That test pattern means extracted features of that word used for the testing. In this way the test pattern is tested against the training vector by using various classifier such as SVM, KNN, HMM and ANN [6] etc. This classifier classifies the pattern. If the testing word pattern matches with the training pattern class then it means that particular pattern is recognized from training phase and that corresponding pattern is displayed as the output. The system block diagram of speech recognition process is shown in Fig. 1.

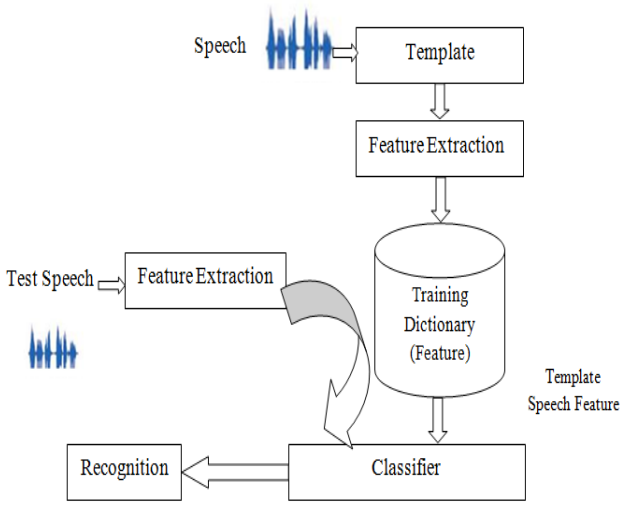

Fig 1: Working Of Speech Recognition Process

II. METHODOLOGY

A. *Feature Extraction*

Feature extraction step finds the set of parameters of utterances that have acoustic correlation with speech signals and these parameters are computed through processing of the acoustic waveform [7]. Feature extraction that creates acoustic observation vectors [8]. The extraction of the best parametric representation of acoustic signals is an important

task to produce a better recognition performance. These features are important for the next phase since it affects behavior of speech recognition system.

B. MFCC in speech recognition

The mel-frequency cepstrum (MFC) is a representation of the short-term power spectrum of a sound, based on a linear cosine transform of a log power spectrum on a nonlinear Mel scale of frequency [9].

It is popular feature Extraction technique [10].Mel-frequency cepstral coefficients are the feature that collectively makes Mel-frequency cepstral (MFC) [11]. The difference between the cepstrum and the mel-frequency cepstrum is that in Mel-frequency cepstral (MFC), the frequency bands are equally spaced on the Mel scale, this mean that it approximates the human auditory system's response more firmly than the linearly-spaced frequency bands used in the normal cepstrum. This frequency warping allows better representation of sound [12].

Sounds generated by a human are filtered by the shape of the vocal tract etc. This shape determines what sound comes out. If we succeed to determine the shape accurately, this gives us an accurate representation of the phoneme being produced. The shape of the vocal tract manifests itself in the envelope of the short time power spectrum, and the function of MFCCs is to accurately represent this envelope. Mel Frequency Cepstral Coefficients (MFCCs) are a feature widely used in automatic speech and speaker recognition. MFCC introduced by Davis and Mermelstein in the 1980's [13].

MFCC is based on human hearing perceptions which cannot perceive frequencies over 1Khz. In other words, in MFCC is based on known variation of the human ear's critical bandwidth with frequency. It has two types of filter which are spaced linearly at low frequency below 1000 Hz and logarithmic spacing above 1000Hz [14]. Pitch is used on Mel Frequency Scale to capture important characteristic of speech. MFCC takes human perception sensitivity with respect to frequencies into consideration so MFCC is best for speech recognition. The block diagram of MFCC as given in is shown in Fig 2:

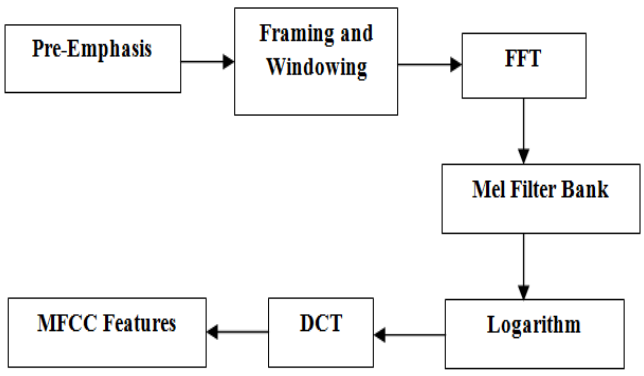

Fig 2: Block Diagram of MFCC

As shown in Figure 2, MFCC consists of seven computational steps. Each discussed briefly in the following:

1) Pre–emphasis

The goal of pre-emphasis is to compensate the high-frequency part that was suppressed during the sound production mechanism of humans. It can also add the importance of high-frequency formants.

The speech signal s (n) is sent to a high-pass filter [15]:

$$s_2(n) = s(n) - a*s(n-1)$$

Where s_2 (n) is the output signal and the value of a is generally between 0.9 and 1.0 and z-transform of the filter is

$$H(z) = 1 - a*z^{-1}$$

2) Framing

An audio signal constantly changes, so to simplify this we assume that on short time scales the audio signal doesn't change much (signal doesn't change means statistically i.e. statistically stationary, samples changes constantly on even short time scales). This is why we have to frame the signal into 20-40ms frames. If the frame is much shorter we don't get enough samples to have a reliable spectral estimation and if it is longer the signal changes highly throughout the frame.

The input speech signal is segmented into frames of 20~30 ms with optional overlap of 1/3~1/2 of the frame size. Usually the frame size (in terms of sample points) is equal to power of two in order to facilitate the use of FFT [16]. If this is not the, we need to do zero padding to the nearest length of power of two. If the sample rate is 16000Hz and the frame size is 480 sample points, then the frame duration is 480/16000 = 0.03 sec = 20 ms. Additional, if the overlap is 160 points, then the frame rate is 16000/ (480-160) = 50 frames per second.

The process of segmenting the speech samples obtained from analog to digital conversion (ADC) into a small frame with the length within the range of 20 to 40 msec. The speech signal is divided into N samples of frames. An Adjacent frames are being separated by M (M<N).

Typical values used are M = 100 and N= 256.

3) Windowing

The important function of Windowing is to reduce the aliasing effect, when cut the long signal to a short-time signal in frequency domain [17].

Different types of windowing functions:

- Rectangular window
- Bartlett window
- Hamming window

Out of these, the most widely used window is Hamming window. Hamming window is used as window shape by considering the next block in feature extraction processing chain and integrates all the closest frequency lines. Each frame has to be multiplied with a hamming window in order to keep the continuity of the first and the last points in the frame (to be detailed in the next step). If the signal of frame is denoted by s (n), n = 0…N-1, then the signal after Hamming windowing is s (n)*w (n), where w (n) is the Hamming window defined by:

$$W(n, a) = (1 - a) - a\cos(2pn/(N-1)), \quad 0 \leqq n \leqq N-1$$

4) Fast Fourier Transform

Spectral analysis shows that different accents in speech signals correspond to different energy distribution over frequencies. Therefore we perform FFT to obtain the magnitude frequency response of each frame [18].

When we implement FFT on a frame, we consider that the signal in frame is periodic, and continuous. If this is not the case though, we can perform FFT but the discontinuity at the frame's first and last points is likely to introduce undesirable effects in the frequency response. To handle this problem, we have two strategies [19]:

a. Multiply each frame by a Hamming window to increase its continuity at the first and last points.

b. Take a frame of a variable size such that it always contains an integer multiple number of the fundamental periods of the speech signal.

Practically the second strategy encounters difficulty because the identification of the fundamental period is not a minor problem. Moreover, unvoiced sounds do not have a fundamental period at all. We generally adopt the first strategy to multiply the frame by a Hamming window before performing FFT [20].

5) Mel Filter Bank

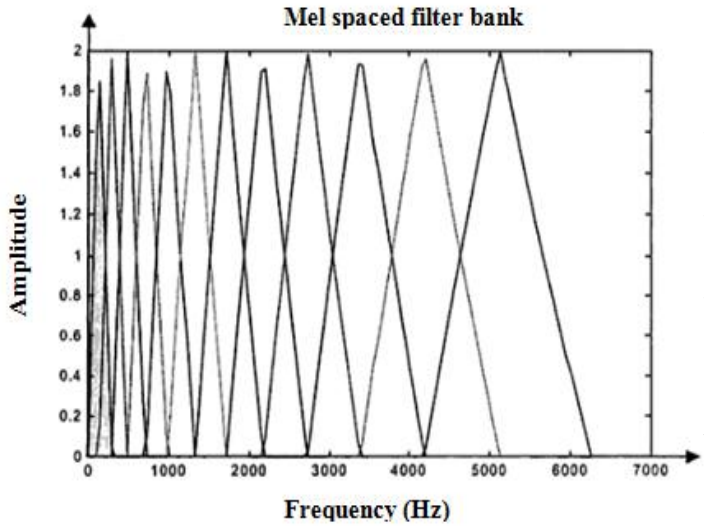

Fig 3: An Example of Mel-Spaced Filter Bank

Triangular Bandpass Filters are used because the frequency range in FFT spectrum is very wide and voice signal does not follow the linear scale. We multiple the magnitude frequency response by a set of 20 triangular bandpass filters to get the log energy of each triangular bandpass filter. The position of these filters is equally spaced according to the Mel frequency, which is related to the linear frequency f by the following equation [21]:

Mel (f) =1125*ln (1+f/700)

The Mel-frequency is proportional to the logarithm of the linear frequency, reflecting similar effects in the human's subjective aural perception.

Advantages of triangular bandpass filters:

- Smooth the magnitude spectrum such that the harmonics are flattened in order to obtain the envelope of the spectrum with harmonics. This suggests that the pitch of a speech signal is generally not presented in MFCC.
- Reduce the size of the features tangled in it.

6) Discrete Cosine Transform

This is the process to convert the log Mel spectrum into time domain using Discrete Cosine Transform (DCT) [22]. In this step, we apply DCT on the 20 log energy E_k obtained from the triangular bandpass filters to have L mel-scale cepstral coefficients. Formula for DCT is as shown in below:

$$C_m = S_{k=1}^{N} \cos [m*(k-0.5)*p/N]*E_k, \quad m=1, 2... L$$

Where N is the number of triangular bandpass filters and L is the number of mel-scale cepstral coefficients. Generally we set N=20 and L=12.

7) MFCC Features

In this way Mel-frequency cepstral coefficients are extracted from the speech signal. These features are the main component of speech recognition process. Further classification of these features is done by the various types of Classifier.

C. KNN CLASSIFIER

KNN is Instance-based classifier [23]. Instance-based classifier performs on the premises that classification of unknown instances can be done by relating the unknown to the known according to some distance/similarity function. Opinion is that two instances far apart in the instance space defined by the appropriate distance function are less likely than two closely situated instances to belong to the same class.

1) The learning process

Instance-based learners do not abstract any information from the training data during the learning phase. Learning is simply a question of encapsulating the training data. The process of generalization is deferred until it is absolutely inevitable, that is, at the time of classification. Due to this property instance-based learners referred as lazy learners [24].

2) Classification

Classification (generalization) using an instance-based classifier can be a simple matter of locating the nearest neighbor in instance space and labeling the unknown instance with the same class label as that of the located (known) neighbor. This approach is often referred to as a nearest neighbor classifier. The weakness of this simple approach is the lack of robustness that characterizes the resulting classifiers. The high amount of local sensitivity makes nearest neighbor classifiers highly susceptible to noise in the training data [25].

More robust models can be achieved by locating k, where k > 1, neighbor and letting the majority vote decide the outcome of the class labeling. A higher value of k results in a smoother, less local sensitive, function. Nearest neighbor classifiers can be considered as a special case of the more general k-nearest neighbor's classifier so referred as a KNN classifier. The disadvantage of increasing the value of k is that as k approaches n and n is the size of the instance base. The performance of the classifier can come up to the most straightforward statistical baseline, the conclusion that all unknown instances belong to the class most frequently represented in the training data.

This problem can be avoided by limiting the influence of distant instances. This can be done by assigning a weight to each vote and Weight is a function of the distance between the unknown and the known instance. Each weight can be defined by the inversed squared distance between the known and unknown instances votes cast by distant instances will have very little influence on the decision process compared to instances in the near neighborhood. Distance weighted voting generally serves as a good middle ground as far as local sensitivity is concerned.

III. DATABASE PREPARATION

For the development of this system, we first prepare the database .The database has 90 agricultural related hindi isolated words such as chana, masur, mung, rajma, Arhar, genhu, chaval, ganna, ragi and til etc. These words were recorded by praat software [26] from ten Hindi speakers at 16KHZ. 8 speakers age were between 20 to 30 and 2 speaker age was about 45-52. Recording were done in room environment. Each word is uttered 3 times. Out of these 2 utterances of every word were selected as training data and 1 utterance selected as testing data for such speaker dependent system. Hence our training data was of 1600 wav files and test data of 800 wav files. All recording is done by Sennheiser PC360 Headset. The PC360 headset has noise cancellation facility and the signal to noise ratio (SNR) is less [27].

IV. EXPERIMENT RESULTS

Experiment had done on two types of database:

- Experiment No.1 was done on First 10 words.
- Experiment No.2 was done on all 80 words.

A. *First Eexperiment*

In Experiment No. 1 we have used 10 Hindi words, 10 speaker and 3 utterances of each words. As we mentioned first 2 utterances of each word were added in training data and 1 utterance is added in testing data. Hence total training data was 0f 200 wav files and testing data was of 100 wav files.12th order MFCC features were used for training and test data. Both training and testing data features were applied on KNN classifier. As there were 10 words 10 classes were created. According to classes respective wave file had been classified by the KNN. After classification we got overall rate of classification as 89%. This system runs quickly. Rate of classification of each word tells how much percent that word had been matched with its respective class. Rate of classification of all ten words is shown on the fallowing graph Fig.4:

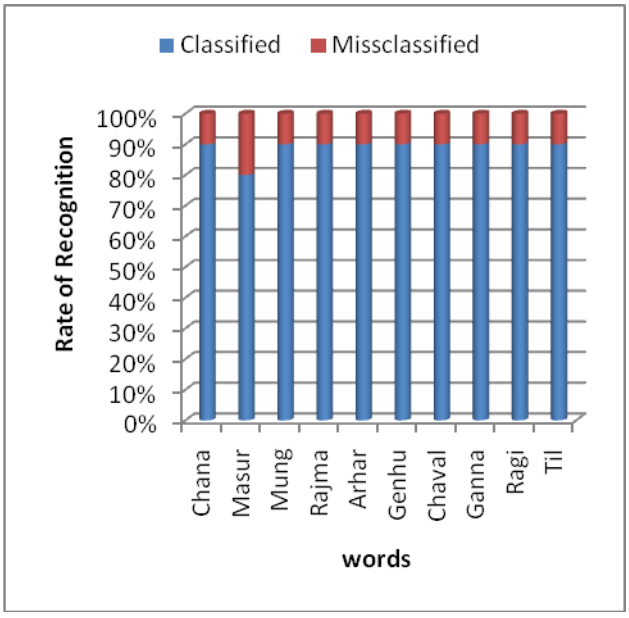

Fig 4: Graphical representation of classification of 10 Hindi words

B. *Second Experiment*

In second experiment we have used 80 Hindi words, 10 speaker and 3 utterances of each words. We were having 1600 wav file in training data and 800 wave files in testing data. Using 12 the order MFCC features were extracted from all 1600+800=2400 wav files. This training and testing data were applied on KNN classifier. KNN classifier has given total recognition rate 62.50%. This increased database system takes time to run.

C. *Comparison of experiment*

TABLE I. COMPARISON OF EXPERIMENT NO.1 AND EXPERIMENT NO.2

Sr. No	Factors	Experiment No.1	Experiment No.2
1	Number of word	10	80
2	Number of speaker	10	10
3	Number of utterances	3	3
4	Training Data	200	1600
5	Testing Data	100	800
6	Total corpus	300	2400
7	Vocabulary Size	Medium	Large
8	Rate of recognition	89%	62.50%
9	Time required to run	less	More
10	Feature Extraction	MFCC	MFCC
11	Classifier	KNN	KNN

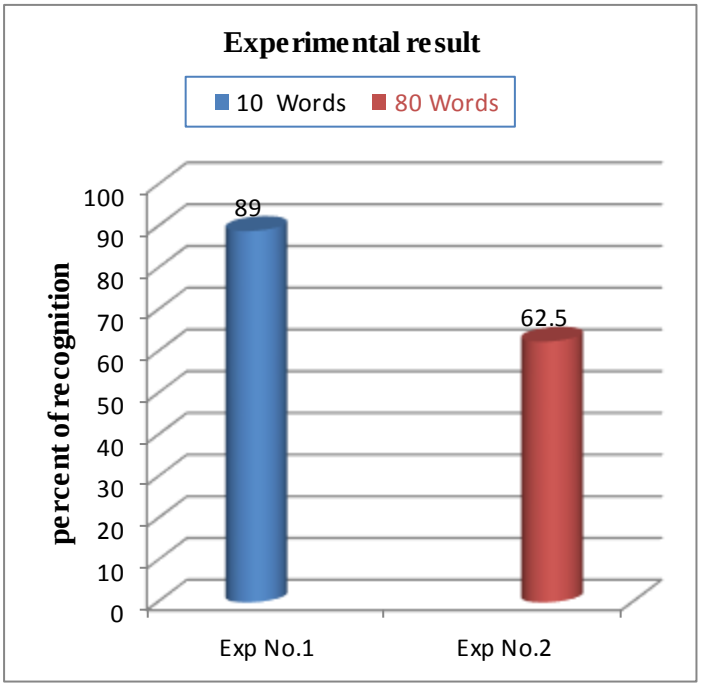

Fig 5: Graphical representation of both results

After comparing Experiment No.1 and Experiment No.2 result we came to the conclusion that as database size increases the recognition rate get decreases. Because every wave file has different numbers of features. We gather all types of features at specific range so as to pass it to the classifier some features of wave file lost. Because classifier requires same number of column features. So these lost features may be of important to that particular word. As features lost the rate of classification decreases and because of that performance of second experiment decreased which has large vocabulary size.

V CONCLUSION AND FUTURE WORK

In this paper, we have briefly discussed MFCC as feature extraction technique. Using MFCC and KNN we developed our Isolated Word Recognition System for Hindi Language. MFCC and KNN have given us 89% of recognition rate for 300 vocabulary data which is better than the 16000 vocabulary size of second experiment. Hence as the size of vocabulary increases rate of classification decreases and due to this our system has given poor performance at large vocabulary size. Further ANN classifier can be used and also speaker independent system can be developed.

ACKNOWLEDGMENT

This work is supported by University Grants Commission under the scheme Major Research Project entitled as "Development of Database and Automatic Recognition System for Continuous Marathi Spoken Language for agriculture purpose in Marathwada Region". The authors would also like to thank the Dept of CS&IT, Dr.Babasaheb Ambedkar Marathwada University Authorities for providing the infrastructure to carry out the research.

REFERENCES

[1] Malay Kumar, R K Aggarwal, Gaurav Leekha and Yogesh Kumar "Ensemble Feature Extraction Modules for Improved Hindi Speech Recognition System", International Journal of Computer Science Issues, Vol. 9, Issue 3, No 1, May 2012.

[2] Pukhraj P. Shrishrimal, Vishal B. Waghmare, Ratnadeep Deshmukh, "Indian Language Speech Database: A Review", I international Journal of Computer Application, Vol 47– No.5, June 2012.

[3] Hemakumar, Punitha, "Speech Recognition Technology: A Survey on Indian languages", International Journal of Information Science and Intelligent System, Vol. 2, No.4, 2013.

[4] Santosh V. Chapaneri , "Spoken Digits Recognition using Weighted MFCC and Improved Features for Dynamic Time Warping", International Journal of Computer Applications, Vol. 40– No.3, February 2012.

[5] M. A. Anusuya, S.K. Katti, "Speech Recognition by Machine: A Review", (IJCSIS) International Journal of Computer Science and Information Security, Vol. 6, No. 3, 2009.

[6] Abhishek Thakur, Naveen Kumar, "Automatic Speech Recognition System for Hindi Utterance with Regional Indian Accents: A Review", International Journal of Electronics & Communication Technology, Vol. 4, April – June 2013.

[7] Preeti Saini, Parneet Kaur, Mohit Dua, "Hindi Automatic Speech Recognition Using HTK", International Journal of Engineering Trends and Technology (IJETT) – Vol. 4, June 2013.

[8] R.K. Aggarwal · M. Dave, "Integration of multiple acoustic and language models for improved Hindi speech recognition system", Springer Science+Business Media, LLC 2012.

[9] Louis-Marie Aubert, Roger Woods, Scott Fischaber, and Richard Veitch "Optimization of Weighted Finite State Transducer for Speech Recognition", IEEE Transactions on Computers, Vol. 62, No. 8, August 2013.

[10] Ankit Kumar, Mohit Dua, Tripti Choudhary, "Continuous Hindi Speech Recognition Using Monophone based Acoustic Modeling", International Journal of Computer Applications 2014.

[11] S B Harisha , S Amarappa , Dr. S V Sathyanarayana, "Automatic Speech Recognition - A Literature Survey on Indian languages and Ground Work for Isolated Kannada Digit Recognition using MFCC and ANN", International Journal of Electronics and Computer Science Engineering.

[12] Borde, Prashant, Amarsinh Varpe, Ramesh Manza, and Pravin Yannawar. "Recognition of isolated words using Zernike and MFCC features for audio visual speech recognition." International Journal of Speech Technology.2015.

[13] Suman K. Saksamudre, P.P. Shrishrimal, R.R. Deshmukh "A Review on Different Approaches for Speech Recognition System", International Journal of Computer Applications, Vol115 – No. 22, April 2015.

[14] Mel-frequency cepstrum, Wikipedia, accessed 26 June 2015.

[15] Anand Vardhan Bhalla, Shailesh Khaparkar "Performance Improvement of Speaker Recognition System", International Journal of Advanced Research in Computer Science and Software Engineering, Vol 2, March 2012.

[16] Munish Bhatia1, Navpreet Singh2, Amitpal Singh, "Speaker Accent Recognition by MFCC Using KNearest Neighbour Algorithm: A Different Approach", International Journal of Advanced Research in Computer and Communication Engineering, Vol. 4, Issue 1, January 2015

[17] Mel Frequency Cepstral Coefficient (MFCC) tutorial, Accessed 24 June 2015.

[18] Divyesh S.Mistry, Prof.A.V.Kulkarni, "Overview: Speech Recognition Technology, Mel frequency Cepstral Coefficients (MFCC), Artificial Neural Network (ANN)", International Journal of Engineering Research & Technology (IJERT) Vol. 2 Issue 10, October – 2013.

[19] Shivanker Dev Dhingra, Geeta Nijhawan, Poonam Pandit, "Isolated Speech Recognition Using MFCC And DTW", International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering, Vol. 2, August 2013.

[20] Shikha Gupta1, Jafreezal Jaafar, Wan Fatimah wan Ahmad, Arpit Bansal "Feature Extraction Using Mfcc", Signal & Image Processing : An International Journal (SIPIJ) Vol.4, No.4, August 2013.

[21] Rajesh Kumar Aggarwal, "Improving Hindi Speech Recognition Using Filter Bank Optimization and Acoustic Model Refinement"PHD Thesis, 2012.

[22] M. Kalamani, Dr. S. Valarmathy, S. Anitha , "Automatic Speech Recognition using ELM and KNN Classifiers", International Journal of Innovative Research in Computer and Communication Engineering, Vol. 3, Issue 4, April 2015.

[23] Tsang-Long Pao, Wen-Yuan Liao and Yu-Te Chen, "A Weighted Discrete KNN Method for Mandarin Speech and Emotion Recognition", International Journal of Innovative Computing, Information and Control ICIC International Volume 6, February 2010.

[24] http://www.fon.hum.uva.nl/praat/manual/kNN_classifiers_1__What_is_a_kNN_classifier_.html, accessed 25 June 2015.

[25] Tsang-Long Pao, Wen-Yuan Liao and Yu-Te Chen, "A Weighted Discrete KNN Method for Mandarin Speech and Emotion Recognition", www.intechopen.com.

[26] Wiqas Ghai, Navdeep Singh, "Literature Review on Automatic Speech Recognition", International Journal of Computer Applications (0975 – 8887) Volume 41– No.8, March 2012.

[27] S.B. Magre, R.R. Deshmukh, "A Review on Feature Extraction and Noise Reduction Technique", International Journal of Advanced Research in Computer Science and Software Engineering Vol 4, Issue 2, February 2014.