Volume 5, Issue 8, August 2015 ISSN: 2277 128X

International Journal of Advanced Research in Computer Science and Software Engineering

Research Paper

Available online at: www.ijarcsse.com

Speech Recognition System for Punjabi Language

Harpreet Kaur, Rekha Bhatia

Department of Computer Science, Punjabi University,

Patiala, Punjab, India

Abstract – Speech Recognition has been a wide area of research for a long time now. Researchers have been putting a lot of efforts and devised different methods for the same. Spoken Punjabi alphabets is one of the subsets of the speech recognition technique. It has many applications. Though, this task seems to be simple, however it is one of the most difficult tasks in speech recognition due to highly confusing set of alphabets that have much similar pronunciation. This paper aims at developing a speech recognition system for the set of Punjabi alphabets using Mel Cepstrum Frequency Coefficients (MFCCs) for feature extraction. The system has been trained using Hidden Markov Model (HMM) for estimating the parameter. Testing has been done using the Viterbi Algorithm and the results are calculated in three scenarios. The accuracy obtained is 80%, 100% and 55% respectively.

Keywords: HMM, MFCC, Viterbi Algorithm, MATLAB, Punjabi language.

I. INTRODUCTION

Speech recognition technique has been explored widely for the past four decades as the degree of acceptance for such systems is high. Speech is considered to be one of the easiest and comfortable means of communication. It is an efficient way to recognize a person on the basis of speech. It is an important biometric authentication process [32]. To exchange any information between man and machine keyboards, pointing devices etc are required which is not convenient for a layman as it requires special skills. Hence, speech has provided a great platform to resolve this issue. Speech recognition is the process where the human speech is input into the system in analog form and the machine converts it into digital form to make it understandable. Many such systems have been developed ad worked upon to maximize the ease of access for communication.

Speech recognition systems are further classified into speaker verification systems and speaker identification systems where it identifies the speaker's claimed identity is verified and the speakers itself is verified. Also, a class of speech recognition deals with speaker-dependent and speaker-independent systems. The development workflow for speech recognition follows the acquisition of speech, feature extraction module, the recognition model and the testing module. In this work, implementation is done using MATLAB. MFCCs are used for feature extraction because it is designed using the knowledge of human auditory system and is used in every state of speech recognition system or art speech. It is used mostly as it is believed to mimic the behavior of human ear [34]. HMM is one of the popular statistical tools for modeling a wide range of time series data. In speech classification for speech recognition, HMM has been proved to be of great success [14]. Hence, for training HMMs are used. To test the system, Viterbi Algorithm is implemented which is a dynamic programming and decoding algorithm that allows the computation of the most probable path.

A lot of research has been carried out in this field for different languages because of its wide area of application like, automated directory assistance to retrieve information such as spelling names, telephone numbers addresses [5]. Various techniques have been implemented for isolated words [25], connected words, continuous speech for small vocabulary using the HTK toolkit on Linux platform [10]. However, the need of hour is to generate a system on Windows platform to maximize the ease of access [10], [15].

Spoken alphabet recognition is done in this work. The task may seem easy but for machines it can be challenging task due to high acoustic similarities among certain group of letters. The alphabets like ਕ, ਖ, ਗ, ਘ, ਚ have high acoustic similarities which makes it confusing for the system to recognize. Hence, a technique has been developed for the recognition of such confusing set of letters.

II. SYSTEM ARCHITECTURE

The proposed algorithm follows three steps. In the first step, preprocessing of the data is done where the voice signal is converted from analog to digital, i.e. sampling is done. The second step proceeds with calculating the parameters that will be used for recognition purpose. This phase involves feature extraction and is referred to as training module. In the last step, the process of feature matching which involves the testing procedure is carried out. This module is known as testing module.

The following figure shows the steps carried out for the proposed system. In the following sections of this chapter, the detail about each of these steps is discussed including the implementation procedure.

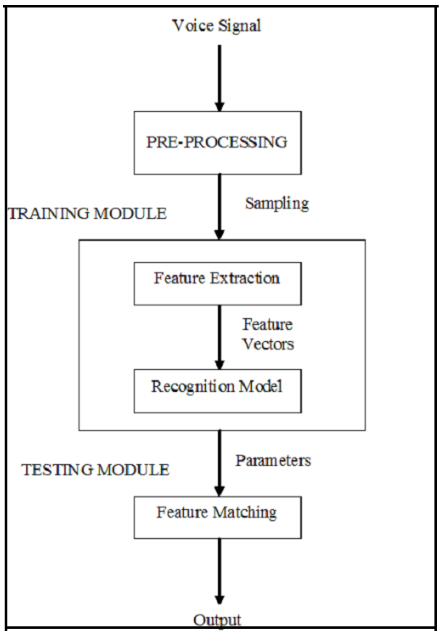

Fig. 1 Design of the proposed system

A. Pre-processing

The foremost step is the generation of the speech data corpus which is followed by the preprocessing phase, however, the speech signal captured decides the recognition accuracy base on the microphone's quality. Analog signal has a number of variable frequency components.

In this phase, the analog signals captured using a microphone is digitized according to the Nyquist theorem, which says that the signal must be sampled more than twice the highest frequency. For normal microphones, sampling rate of 16 KHz or more is preferred [24]. The analog signal cannot be directly applied in the computer as it understands only digital data. It is necessary to sample the analog signal into the discrete-time signal, which the computer can use to process. The sampled data is then further fed into the training module for processing where feature extraction is done.

B. Feature Extraction

In this phase, the sampled data speech signal is processed further for feature extraction. This generates feature vectors that are used by the recognition model for training the system. A number of feature extraction techniques are available; however Mel Frequency Cepstrum Coefficients have been used here. In automatic speech recognition, using 20 MFCC coefficients is very common but 10-12 coefficients are considered to be enough for encoding speech [34]. The procedure for feature extraction is as follows:

1. **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 [21]. This process will increase the energy of signal at higher frequency [8]. The first order FIR filter is represented as follows:

   $H(z) = 1 - \alpha z^{-1}, \; 0.9 \le \alpha \le 1.0$ (3)

2. **Framing:** In this step, the speech samples are segmented into small frame size within the length of 20 to 40msec. The number of samples used for each frame is 256. To calculate the number of frames, the total numbers of samples in the input voice file are divided by 128.
3. **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. Hamming window is used as it introduces least amount of distortion [21]. This implementation uses Hamming window of length 256. The following function represents the Hamming window function:

   $$h[n] = \begin{cases} 0.54 - 0.46 \cos(2\pi n / N), 0 \le n \le N-1; \\ 0, otherwise \end{cases} \quad (4)$$

4. **DFT Computation:** DFT is the next step, which converts each frame into frequency domain from time domain. It is done to speed up the processing [8]. It is represented by the following equation:

$$X(k) = \sum_{n=0}^{N-1} x(n)\; e^{-\frac{j2\pi kn}{N}}, 0 \le k \le N-1 \quad (5)$$

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 [21].

5. **Mel Frequency Filter Bank:** The Mel frequency filter bank is applied to the Fourier transformed frame obtained from each sample. Since Mel scale helps to space windows equally, the design and implementation becomes easier [38]. 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.

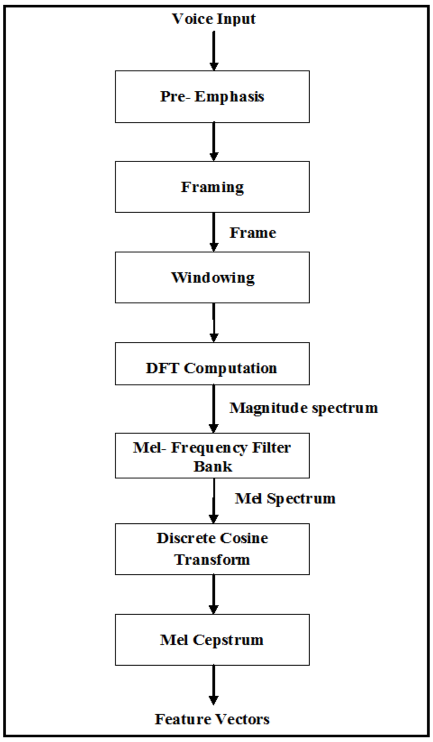

Fig. 2 MFCC feature extraction

Hence, the filter bank is designed to emphasize the low frequency over the high frequency [21]. The following relation defines the relation between frequency of speech and Mel scale, it converts linear scale frequency into Mel scale frequency.

Mel (f) = 2595*$\log_{10}$(1+f/700) (6)

In this work, the number of filters used is filter bank are 32. A total of 42 MFCC parameters include twelve original, twelve delta (First order derivative), twelve delta-delta (Second order derivative), three log energy and three 0th parameter. In this work, 12 coefficients are extracted and used for analysis.

6. **Discrete Cosine Transform:** This is the final step in feature extraction that leads to the generation of feature vectors. The conversion of log Mel spectrum into time domain using Discrete Cosine Transform (DCT) is carried out in this step. The result of the conversion is called Mel Frequency Cepstrum Coefficient. The set of coefficient is called feature vectors. Therefore, each input utterance is transformed into a sequence of feature vector.

C. Hidden Markov Model

Researchers have developed statistical pattern matching techniques such as HMM and GMM. These methods have offered great improvement in ASR by using probability distribution density. Based on these probabilities, the models are created with the entire data for each speech patterns [43]. In this work, Gaussian Mixture HMMs have been used to calculate the likelihood of observation vectors.

An HMM is characterized by the following parameters [30], [31], [39]:

1. The number of states, N, in the model. The individual states $S = \{S_1, S_2, S_3, \ldots\ldots, S_n\}$ and the state at time t as q_1.
2. The number of distinct observation symbols, M, per state, i.e. the discrete alphabet size. The individual symbols are denoted as $V = \{V1, V2, \ldots, Vm\}$
3. The state transition probability distribution $A = \{a_{ij}\}$, where , each a_{ij} is the transition probability from state S_i to S_j. Clearly, $a_{ij} \geq 0$ and $\sum_k aij = 1, \ \forall i$
4. The observation symbol probability distribution $B = b_{jk}$, where each b_{jk} is the observation symbol probability for symbol v_k, when the system is in state S_j. Clearly, $b_{ij} \geq 0$, $\forall j,k$ and $\sum_k bjk = 1, \forall j$.
5. The initial state distribution $\pi = \{\pi\}$ where, $\pi = P[q1 = S1]$, $1 \leq j \leq N$. HMM model can be specified as λ = (A,B, π,M,N,V). In this thesis, HMM is represented as λ = (A, B, π) and assume M, N and V to be implicit.

Three fundamental problems of HMMs are:

1. **Probability evaluation:** Evaluation is to find probability of generation of a given observation sequence by a given model. The recognition result will be the speech unit corresponding to the model that best matches among the different competing models.
2. **Determination of the best sequence:** To find the best possible sequence of states from the given observations.
3. **Learning:** Learning is to adjust the model parameters (A, B, π) to maximize the probability of the observation sequence given the model. It is the most difficult task of the Hidden Markov Modeling, as there is no known analytical method to solve for the parameters in a maximum likelihood model. Instead, an iterative procedure should be used. Baum-Welch algorithm is the extensively used iterative procedure for choosing the model parameters. In

this method, start with some initial estimates of the model parameters and modify the model parameters to maximize the training observation sequence in an iterative manner till the model parameters reach a critical value.

The probability evaluation can be realized easily with the forward algorithm. The determination of the best state sequence is often referred as a decoding or search process and uses Viterbi search algorithm [30].

D. Viterbi Algorithm

The Viterbi Algorithm is often looked upon as minimizing the error probability by comparing the likelihoods of a set of possible state transitions that can occur, and deciding which of these has the highest probability of occurrence. The VA can be simply described as an algorithm which finds the most likely path through a trellis, i.e. shortest path, given a set of observations. The trellis in this case represents a graph of a finite set of states from a Finite States Machine (FSM) [50]. The inputs to this module are initial probability, transition probability, Gaussian mixture, number of HMM states and the output is the state sequence and the output probability. It calculates the coefficients for every speech sound which are matched with those generated by the training module.

III. IMPLEMENTATION

The proposed work is implemented in MATLAB® 2010b using the toolbox VOICEBOX which is externally added into it. Signal Processing Toolbox is used for pre-processing, i.e., filtering and Fourier transform. The system is developed on Windows 8.1 64-bit operating system having Intel i3 (2.40 GHz) processor and 3 GB RAM. The system is designed to recognize the Punjabi alphabets.

A. Speech Corpus

The speech database has been recorded in a noise free room environment using the recording tool *Audacity 2.1.0* which is it latest version and the files are stored in *.wav* format. The speech data is recorded using a unidirectional microphone by keeping an approximate distance of 5-10cm between the mouth and the microphone and no noise reduction mechanism has been applied to the speech files. Sampling rate used for recording is 32 KHz on mono- channel. A total of 9 speakers recorded the data of which 5 are female and 4 are male. For training, the set of 5 speakers are used, 2 females + 3 males while for testing, 4 speakers, 3 females + 1 male speakers voices are used. The age of speakers is between 20 to 50 years. The data set consists of 10 Punjabi alphabets uttered once by every user. 10*9 which gives a total of 90 speech files. Experiment is conducted using these speech files in different proportions. In the Punjabi language, a total of 35 alphabets are there. The following table gives the details about that.

B. Pre-processing

The speech files are sampled at a sampling frequency of 32 KHz where each file has a length of 1 to 2 sec. However, this much length is too large to be analyzed by the system. Hence, framing is done. Usually it takes in a frame of the speech signal every 20-40 msec and performs certain spectral analysis [8]. The steps followed in this module are discussed below:

1. The speech file is read from the directory using the command *'wavread'*.
2. Next step is to determine the length of the speech signal using the *'plot'* command which plots the graph of the speech signal.
3. Now the start and end of the wave is determined and silence is removed from the sound, keeping only the uttered alphabet to increase the accuracy.
4. Using the command *'wavwrite'*, a new speech file is created.
5. The above steps are repeated for all the speech files before proceeding further.

C. Training module

A speech signal cannot be directly fed into the system for analysis. It has to be represented in a more efficient and compact form. The original speech signal is converted into a series of feature vectors using feature extraction technique MFCC.

MFCC performs a series of steps to generate the feature vectors that are used for training the system. The following procedure is implemented:

1. Pre-emphasis follows the process of filtering. The FIR filter is used to flatten the speech signal. The number of filters used by this system is 32. The command *'filter'* is used here.
2. Framing is the next step in this procedure where the speech signal is divided into a number of frames.
3. Windowing is done to minimize the discontinuities in the frames. Each frame is multiplied by a windowing function.
4. After the windowing, Fast Fourier Transformation (FFT) is calculated for each frame to extract frequency components of a signal in the time-domain.
5. The above step is used to calculate Mel Filter Bank which generates the mel spectrum coefficients that are further converted into time domain as mel spectrum coefficients are real numbers using the DCT procedure.
6. The final step is the generation of Mel Frequency Cepstrum Coefficients. In this work, 12 coefficients are obtained for each frame. The command used for feature extraction is *'melcepst'*. A matrix is obtained with the number of frames as rows and mel cepstrum coefficients as columns.

D. Recognition Model

The system is trained using Hidden Markov Model. It is statistical approach. HMM initialization is done using a prototype model. The HMM prototype gives the model topology specification, number of states used, transition parameters and the output distribution parameters. The number of states varies according to the input data.

This work uses 4 states for HMMs, namely, number of Baum-Welch re-estimation parameters, i.e., transition parameters are set to 5 and the Gaussian Mixture model parameters are set to 3. Parameters are re-estimated repeatedly until the re-estimation for the training data converges.

E. Testing Module

This is the last phase in speech recognition which performs feature matching, i.e., to find the most probable sequence that matches the outputs obtained from the training module. Once the system model is generated, it is used for the recognition of an unknown utterance, this is known as testing. Viterbi algorithm is used for this purpose.

The Viterbi algorithm is a dynamic programming decoding algorithm that allows the computation of the most probable path. The inputs to this module are initial probability, transition probability, Gaussian mixture, number of HMM states and the output is the state sequence and the output probability.

As in the training module, feature vectors are generated for training the system. Likewise, in this module as well a set of feature vectors is generated from the test data. The command *'viterbi'* is used for processing the signal which matches it to the recognition model. If two or more transitions are found to be the maximum, i.e. their metrics are the same, and then one of the transitions is chosen randomly as the most likely transition.

IV. EXPERIMENT AND RESULTS

The tests are performed on the speech data recorded using a unidirectional microphone in a noise free room environment. The speech waves have not been filtered using any specific noise reduction technique. The tests are conducted using the speech data of 5 female and 10 male speakers. The data set is divided into two parts, where in the first part the tests are conducted using the first 5 alphabets and the second part consists of the further 5 alphabets of Punjabi language.

For training, the speech data consists of the set of 5 speakers, 2 females + 3 males while for testing, 4 speakers are considered, 3 females + 1 male speaker's voices are used. The recognition rate, i.e., the accuracy is calculated as follows [1]:

$$Recognition\ rate = \frac{Successfully\ \ detected\ words}{Number\ \ of\ words\ in\ test\ set}$$

A. Performance Evaluation Parameters

The speech corpus is divided into two sets, consisting of 45 words each. The accuracy of the system is tested in 3 different scenarios as follows:

1. System trained using the first 5 alphabets of Punjabi language where 5 speakers, 2 female + 3 male speakers are used to train the system. For testing, 4 different speakers, 3 male+ 1 female speaker are used and the tests are conducted. The letters are shown in the table below.

Table 1. Alphabets used in first case

ੳ	ਅ	ੲ	ਸ	ਹ
Ura	Era	Iri	Sussa *Sa*	Haha *Ha*

2. In the second case, the system is trained and tested using the same set of speech data. The first 5 alphabets of Punjabi language are used where 5 speakers, 2 female + 3 male speakers.
3. In the last case, the next 5 alphabets of Punjabi language where 5 speakers, 2 female + 3 male speakers are used to train the system. For testing, 4 different speakers, 3 male+ 1 female speaker are used and the tests are conducted.

Table 2. Alphabets used in the second case

ਕ	ਖ	ਗ	ਘ	ਚ
Kukka *Ka*	Khukha *Kha*	Gugga *Ga*	Ghugga *Gha*	Chucha *Ca*

B. Results

In the proposed method, the implementation is done in MATLAB® 2010b using the toolbox VOICEBOX which is externally added into it. Signal Processing Toolbox and its functions are used for pre-processing and the results are found to be varying from 60% to 97%. The tests are performed on Punjabi alphabets with data collected from 9 speakers, 5 female and 4 male. The data set has been divided into 2 groups and the experiments are performed in three scenarios as described above.

The recognition rate is found to be 80%, 100% and 55% accurate for first, second and third case respectively.

Table 3. Recognition results

S.no.	Scenario	Training set	Testing set	Recognition rate
1.	45 words using first 5 alphabets	3 male + 2 female speakers	1 male + 3 female speakers	80 %
2.	45words using first 5 alphabets	3 male + 2 female speakers	3 male + 2 female speakers	100%
3.	45 words using next 5 alphabets	3 male + 2 female speakers	1 male + 3 female speakers	55%

V. CONCLUSION

An Automatic Speech Recognition system for Punjabi Language has been proposed in this research work. The system so developed is found to be quite accurate and efficient. Though a number of ASRs have been developed for Punjabi Language but most of the work has been done in the Linux platform using the HTK toolkit, however the field has not been explored using Windows platform. The HTK toolkit is known to generate accurate results for all kinds of vocabulary sizes but it is quite complex to learn. The work done in this thesis uses purely MATLAB functions.

The algorithm proposed uses inbuilt MATLAB functions and an additional toolbox VOICEBOX. The system uses the speech data recorded in a noise free room environment. MFCCs are used to extract features from the speech signal. System has been trained using HMM which is considered to be one of the most efficient pattern recognition techniques. Viterbi Algorithm has been used for system testing as it is found to be the one that finds the most probable sequence of path. Hence, the implementation is done using the above mentioned techniques.

The system is proposed for recognition of 90 alphabets taken from Punjabi language with 9 speakers having Punjabi as their native language from the age group of 20-50 years. The accuracy of the system is evaluated in three different scenarios as described in the previous sections. The results are found to be 80%, 100% and 55% accurate for each scenario respectively.

An extension to this work can be done by considering all the 35 alphabets of Punjabi Language. It is a quite challenging task, as training the system is a time consuming process. The major task is to use a large vocabulary size and improve the recognition rate.

The work can also be extended to the use of isolated words and continuous speech not only from the native speakers but from people belonging to different areas. This will contribute significantly to this Automatic Speech Recognition system.

REFERENCES

[1] Aggarwal R. K. Dave, M, "Using Gaussian Mixtures for Hindi Speech Recognition System", in *International Journal of Signal Processing, Image Processing and Pattern Recognition*, vol.4, no.4, pp. 157–170, 2011

[2] Alphabet. T, "An introduction to Gurmukhi", *Punjabi Computing Resource Centre*, 2005 http://guca.sourceforge.net/resources/introductiontogurmukhi/an.introduction.to.gurmukhi.pdf

[3] Amin. T. Bin. Mahmood, I, "Speech Recognition using Dynamic Time Warping", in *2nd International Conference on Advances in Space Technologies*, pp. 74–79, 2008.

[4] Anusuya, M., & Katti, S., "Speech recognition by machine: A review", in *International Journal of Computer Science and Information Security*, vol.6, no.3, pp. 181–205, 2009.

[5] B. Adam. T. Salam, "Spoken English Alphabet Recognition with Mel Frequency Cepstral Coefficients and Back Propagation Neural Networks," *International Journal of Computer Applications* pp. 21–27.

[6] Beg A. Hasnain, S. K, "A speech recognition system for Urdu language BT," Lecture Notes in Computer Science, 118–126, 2008.

[7] Cole. R. Fanty. M. Muthusamy, Y. Gopalakrishnan, M, "Speaker-independent recognition of spoken English letters," *International Joint Conference on Neural Networks*, vol. 2, pp. 45–51, 1990.

[8] Dave. N, "Feature Extraction Methods LPC, PLP and MFCC in Speech Recognition", *International Journal for Advance Research in Engineering and Technology*, vol.1 no.6, pp.1–5, 2013.

[9] Doye, B. a S. D. D," Speech Recognition Using Vector Quantization through Modified K-means LBG Algorithm, *3*(7), pp. 137–145, 2012.

[10] Dua. M. Aggarwal. R. K. Kadyan. V.Dua. S," Punjabi Automatic Speech Recognition Using HTK," *9*(4), 359–364, 2012

[11] El-Ramly. S. H. Abdel-Kader. N. S. El-Adawi. R," Neural networks used for speech recognition," *Proceedings of the Nineteenth National Radio Science Conference*, pp. 1–7, 2002.

[12] Gaikwad, S. K. Gawali, B. W. Yannawar.P, "A Review on Speech Recognition Technique", *International Journal of Computer Applications*, vol.10. no.3, pp.16–24, 2010.

[13] Gaurav, G," Development of Application Specific Continuous Speech Recognition System in Hindi," *Journal of Signal and Information Processing*, vol.3 no.3, pp. 394–401, 2012

[14] Geng, Y. Wang. G. Zhu. C. Fei. T. Liu. X, "Speaker Recognition System Based on VQ in MATLAB", pp. 494–501, 2012

[15] Ghai,W, "Continuous Speech Recognition for Punjabi Language", *International Journal of Computer Applications*", vol.72 no.14, pp. 23–28, 2013.

[16] Ghai, W. Singh, N," Analysis of Automatic Speech Recognition Systems for Indo-Aryan Languages : Punjabi A Case Study", *International Journal of Soft Computing and Engineering (IJSCE)*, vol.2 no.1, pp. 379–385, 2012.

[17] Ghai, W. Singh, N, "Literature Review on Automatic Speech Recognition", *International Journal of Computer Applications*, vol.41 no.8, pp. 42–50, 2012

[18] Grewal, S. S. Kumar. D, "Isolated word recognition system for english language *Pattern Recognition",* vol.2 no.2, pp. 447–450, 2010.

[19] Gupta, R, "Speech Recognition for Hindi", *M.tech Thesis Report in Department of Computer Science and Engineering, Indian Institute of Technology, Bombay*, 2006

[20] Jain, A. Sharma, O. P," Evaluation of MFCC for Speaker Verification on Various Windows", *IEEE International Conference on Recent Advances and Innovations in Engineering,* 2014

[21] Joshi. S. C. Cheeran.. A. N, "MATLAB Based Feature Extraction Using Mel Frequency Cepstrum Coefficients for Automatic Speech Recognition", *International Journal of Science, Engineering and Technology Research*, vol.3 no.6, pp. 1820–1823.

[22] Kaur, E. A. Singh, E. T, "Segmentation of Continuous Punjabi Speech Signal into Syllables", *The World Congress on Engineering and Computer Science (WCECS)*, vol.1, pp. 20–23, 2010.

[23] Kesarkar. M," Feature extraction for speech recognition," *M. Tech. Credit Seminar Report on Electronic Systems*, pp. 1–12, 2003.

[24] Kumar, K. Aggarwal. R. K. Jain. A, "A Hindi speech recognition system for connected words using HTK", *International Journal of Computational Systems Engineering*, vol.1 no.1, pp. 25-32, 2012.

[25] Mehla, R., & Aggarwal, R. K., "Automatic Speech Recognition : A Survey", *Proc. of the Intl. Conf. on Advances In Engineering And Technology*, 2014. doi: 10.15224/ 978-1-63248-028-6-01-08

[26] Muda. L. Begam. M. Elamvazuthi, "Voice Recognition Algorithms using Mel Frequency Cepstral Coefficient (MFCC) and Dynamic Time Warping (DTW) Techniques", *Journal of Computing*, vol.2 no.3, pp. 138–143, 2010.

[27] Nadungodage, T. Weerasinghe, R, "Continuous Sinhala Speech Recognizer", *Conference on Human Language*, pp. 2–5, 2011.

[28] Padhy, B. K., & Sahu, S. K, "Analysis of Speech Recognition Techniques", Department of Electrical Engineering National Institute of Technology, 2009

[29] Pruthi, T., Saksena, S. and Das, P.K, "Swaranjali: Isolated Word Recognition for Hindi Language using VQ and HMM", *International Conference on Multimedia and Processing,* 13-15 August, IIT Mardras, 2000.

[30] Rabiner. L. R, "A tutorial on hidden Markov models and selected applications in speech recognition", *Proceedings of the IEEE*, vol.77 no.2, pp. 257–286, 1989.

[31] Rabiner, L. R. Juang, and B. H, "An introduction to hidden Markov models", *IEEE ASSP Magazine*, (January), pp. 257–286, 1986.

[32] Rath, P. G. S, "REAL TIME SPEAKER RECOGNITION USING MFCC AND VQ", National Institute of Technology, pp. 1–71.

[33] Ravinder, K, "Comparison of HMM and DTW for Isolated Word", *Progress in Pattern Recognition, Image Analysis, Computer Vision, and Applications Lecture Notes in Computer Science*, Vol. 6419, pp. 244–252, 2010.

[34] Saini, P. Kaur, P, "Automatic Speech Recognition: A Review", *International Journal of Engineering Trends and Technology*, vol.4 no.3, pp. 132–136, 2013.

[35] C. By. S. Mittal, S., & Applications, C, "Development of Phonetic Engine for Punjabi Language", *M.Tech Dissertation in Computer Science and Applications, School of Mathematics and Computer Applications, Thapar University, Patiala,* 2014.

[36] Sharma A. & Kaur A, "Automatic Segmentation of Punjabi Speech Signal Using Group Delay", *Global Journal of Computer Science and Technology Software & Data Engineering,* vol.13 no.12, 2013.

[37] Shrawankar, U., & Thakare, V, "Techniques for Feature Extraction in Speech Recognition System : a Comparative Study", *International Journal of Computer Applications in Engineering, Technology and Sciences (IJCAETS)*, 412–418, 2013.

[38] Singh, A. K., Singh, R., & Dwivedi, A, "Mel Frequency Cepstral Coefficients Based Text Independent Automatic Speaker Recognition Using Matlab", *International Conference on Reliability, Optimization and Information Technology*, pp. 524–527, 2014.

[39] Singh, B. Kapur, N.Kaur, P, "Speech Recognition with Hidden Markov Model : A Review", *International Journal of Advanced Research in Computer Science and Software Engineering*, vol.2 no.3, 2012.

[40] Singh, B Kaur, R Devgun, N & Kaur, R, "The process of Feature Extraction in Automatic Speech Recognition System for Computer Machine Interaction with Humans : A Review", *International Journal of Advanced Research in Computer Science and Software Engineering*, vol.2 no.2, pp.1–7, 2012.

[41] Ghai, W. Singh, N., "Phone based acoustic modeling for automatic speech recognition for Punjabi language", *Journal of Speech Sciences*, vol.1 no.3, pp.69–83, 2013.

[42] Turner, P. "An introduction to speech recognition and its application in the health care arena", *Health Informatics Journal*, vol.1 no.3 pp.113–121.1995

[43] Vimala. C. Radha, V., "Suitable Feature Extraction and Speech Recognition Technique for Isolated Tamil Spoken Words", vol.5 no.1, pp.378–383, 2014.

[44] Yang, T., "The Algorithms of Speech Recognition, Programming and Simulating in MATLAB", *Bachelor's Thesis in Electronics*, January 2012.

[45] Gopi, E. S, "*Digital Speech Processing Using Matlab*", pp.73-93, 2014.
[46] Ning, D., "Developing an Isolated Word Recognition System in MATLAB" pp.1–6.http://www.mathworks.com/tagteam/60673_91805v00_WordRecognition_final.pdf
[47] De Wachter. M. Matton.M. Demuynck. K. Wambacq. P. Cools, R. Van Compernolle, D, "Template-based continuous speech recognition", *IEEE Transactions on Audio, Speech and Language Processing,* vol.15 no.4, pp. 1377–1390, 2007.
[48] Lim, C. P., Woo, S. C., Loh, a. S., & Osman, R, "Speech recognition using artificial neural networks", *Proceedings of the First International Conference on Web Information Systems Engineering*, vol.1 no.3, pp. 406–412, 2012.
[49] Singh C., Singh G. Lehal, Sengupta J., Sharma D, Goyal V., "Information Systems for Indian Languages", International Conference Proceedings, Patiala, India, March 9-11, 2011
[50] M. S. Ryan and G. R. Nudd. "The Viterbi Algorithm", 1993 http://ciao.twiki.di.uniroma1.it/pub/NLP/WebHome/TutorialViterbi.pdf

