Article can be accessed online at http://www.publishingindia.com

Voice Text Concurrent Transmission Based on Locale

Jyoti Madan*, Ajmer Singh**

Abstract

Among human beings, speech is considered to be the principal mode of communication as it is natural as well as efficient way of exchanging one's views, thoughts and information with other(s). This paper takes a tour of ASR system where the user can type text on computer screen not by using keyboard but by providing voice input through his android mobile phone.

General Terms: Speech Recognition

Keywords: Speech Recognition System, MFCC, HMM, N-Gram Dataset, LPC, ASR

1. Introduction

Speech is a unique form of audio data. It is a relatively simple and widely studied type of acoustic signal. Speech recognition is the technology through which a computer system is designed to be capable enough to recognize spoken word. The research work presented in this paper uses a mobile phone to provide speech input to a speech recognition system which in turn gets printed as text on the computer screen This paper is divided into two major parts, in the first part an introductory overview of speech recognition system is provided along with a glimpse of some of the recent researches in the field so that the reader gets a background of this field and the second part discuses the research work under consideration in this paper.

2. Types of Speech

There are different types of speech based on vocal sound.

2.1 Isolated Words

Speech Recognizers give the input in the form of vocal sound by providing the waiting time to do processing.

2.2 Connected Words

Connected words are official as input speech through giving vocal sound with minimum pause separately.

2.3 Continuous Speech

It is very difficult to create i.e computer dictation giving to the user.

2.4 Spontaneous Speech

Spontaneous speech takes the recognizer as natural speaker as voice input.

3. Layer Architecture of Automatic Speech Recognition

3.1 Standard Approaches to Large-Vocabulary ASR

It uses statistical learning techniques to determine the maximum-likelihood compression/expansion of the spectrum for each clustered utterance or speaker (often derived from an unsupervised learning algorithm); these approaches are based on an underlying generative model. Another common component is Linear Discriminant Analysis (LDA) or its less constrained cousin, Heteroscedastic

* Deenbandhu Chhotu Ram University of Science & Technology, Murthal, Haryana, India. E-mail: jyoti0038@gmail.com
** Deenbandhu Chhotu Ram University of Science & Technology, Murthal, Haryana, India. E-mail: ajmer.saini@gmail.com

Linear Discriminant Analysis (HLDA), each of which is trained to maximize phonetic discrimination. This layer transforms cepstral features, typically over several past and future acoustic frames, into a new observation sequence for the recognition system. The resulting features are then used to train a large number of Gaussians that are used in combination to generate likelihoods for particular speech sounds in context. The objective functions such as maximum mutual information (MMI) or minimum phone error (MPE) are typically used to train the Gaussian parameters discriminatively. The parameters of this acoustic model are then altered further for testing by incorporating one of several related methods for adaptation, for instance Maximum-Likelihood Linear Regression (MLLR). This was one of the largest reductions shown from any improvement in the systems under test.

3.2 Tandem Approaches

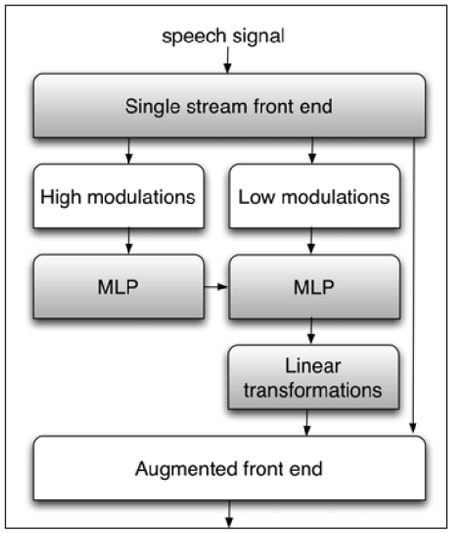
The architectures incorporated more layers; for instance the so-called TRAPS system used such an MLP for a half second of the time sequence of energies for each critical band of the spectrum followed by a combination component that comprised an additional MLP with its own hidden layer. There was no attempt to back propagate errors all the way back through the system. A later form of this system called HATs was trained by taking the input-to-hidden nonlinear transformations from each critical band and using their outputs to feed the final combination MLP.

3.3 Other Approaches

The features were generated by training a large number of Gaussians over the acoustic sequence 2 and computing temporally local posteriors. In practice it provided similar improvements to either MPE training of the acoustic models or to the MLP-based approach described above. Combinations of these methods have also been explored in.

Other approaches have been built on a hierarchical feature approach, for instance training Tandem features for high temporal modulation frequencies and using them, appended to low temporal modulation frequencies as input for a second network generating Tandem features. Thus, one path through the networks encountered four layers of processing by trained parameters while the other encountered two. This method provided significant improvement on a difficult ASR task requiring recognition of speech from meetings as shown in fig 1.

Figure 1. Computational Layers for Hierarchical Modulation Processing



3.4 Comments on Depth and Width

For all of these approaches to improving speech recognition systems via modifications of the observation stream, it often was most natural to incorporate many layers of processing, and to add them on (or insert them) into existing systems without re-engineering the entire structure. Some layers will be trained using an underlying generative model, others will be trained in a purely discriminative manner, while in other cases both approaches will be used. We have typically found that using an insufficient number of units per layer can have a very large effect on the word error rate.

4. Hidden Markov Model: Speech Recognition Technique

HMM is doubly stochastic process with an underlying stochastic process that is not observable, but can only be observed through another set of stochastic processes that produce sequence of observed symbols.

4.1 Description

The Hidden Markov Model (HMM) is a finite state machine i.e said to be a triple *(A, B, II)*.

- π – The initial state distribution.
- A – The state-transition probability matrix.
- B – Observation probability distribution.

4.2 Phases

It is divided into two phases:

(i) Training Phase: speaker voices are recorded and processed in order to generate the model to store in the database.

(ii) Verification Phase: the existing reference templates are compared with the unknown voice input.

4.3 Steps in HMM

1. Evaluation: In this step the probability of a model to generate an observation sequence is judged so as to find the best model available. For the HMM model λ, the observation sequence O is given as :

$$P(O|\lambda)$$

2. Decoding: It is the process wherein the best state sequence, Q, is obtained for the observation sequence, O.
3. Training (Learning): The most tedious step of HMM where the model parameters (A, B, π) are adjusted to maximize the observation sequence probability.

5. Melfrequency Cepstrum Coefficients (MFCC)

The Feature Extraction Technique is used in ASR System as a speech recognizer to render the pitch level of speech i.e done by Mel Frequency Cepstrum Coefficients of Acoustic Model. The aim of this feature extraction process is to obtain a new voice representation which is more compact, less redundant, and more suitable for statistical modeling.

The speech signal follows framing concept in which the voice input being converted into number of sequenced frames. Then, these frames is being transformed through Fourier transform format to warp the formatted frames using frequency values to get the Mel scale. The results get the frequency scale using the logarithmic values and simplified the amplitude and the feature is extracted the compressed frequency information using vector quantization .

The following figure involves the steps of MFCC in fig 2

Figure 2. Steps of MFCC

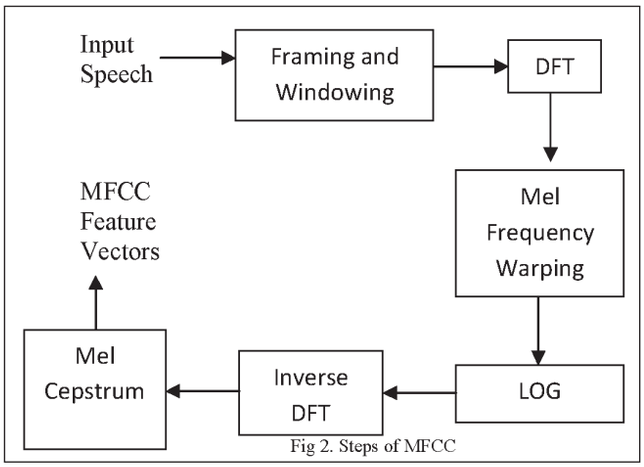


Fig 2. Steps of MFCC

5.1 Framing and Windowing

The speech signal is first decomposed into frames of the size which are usually chosen as a power of two to fit the FFT algorithm. In order to extract the coefficients the speech sample is taken as the input and hamming window is applied to minimize the discontinuities of a signal. The most commonly used window shape is the hamming window

5.2 DFT

Fourier series enable a periodic function to be represented as a sum of sinusoids and converts a speech signal from the time domain to the frequency domain. Then DFT will be used to generate the Mel filter bank.

5.3 Mel Frequency Warping

Perceptual frequency follows the mel scale not the linear for speech signal done by human. Thus, the frequency is measured in Hz and pitch level is measured in mel scale. The major aim of this process is to convert the frequency spectrum to the Mel spectrum. It is a unit of special measure or scale of perceived pitch of a tone. It does not correspond linearly to the normal frequency, but behaves linearly below 1 kHz and logarithmically above 1 kHz.

$$\text{Mel}(f) = 2595*\log10\,(1+f/700) \quad (1)$$

It elaborates the filter bank. The mel scale filter bank is a triangular band pass frequency response and spacing that filtered the scale using auditory system.

It relates to the constant filtering to follow the mel scale with some constant spacing also.

5.4 LOG

It has the effect of changing the multiplication process into the addition process. Therefore, this step converts the multiplication of the magnitude of the Fourier transform into addition.

5.5 Inverse DFT

The speech signal is represented as a convolution between slowly varying vocal tract impulse response and quickly varying glottal pulse. Therefore, by taking the inverse DFT of the logarithm of the magnitude spectrum, the glottal pulse and the impulse response can be separated.

5.6 Mel Cepstrum

In this last step, the compressed frequency information is extracted using the logarithmic values well in time. The result is called the Mel Frequency Cepstrum Coefficients (MFCC). The cepstral representation of the speech spectrum provides a good representation of the local spectral properties of the signal for the given frame analysis. The transform is done to transform the mel coefficient values

$$C_n = \sum_{k=1}^{k} (\log S_k) \cos \{n(k-(1/2)*\pi/k)\} \quad (2)$$

6. Problem Statements and Performance of Systems

Accuracy & speed of processing measures effectiveness. The word error rate (WER), commonly used in ASR assessment, measures the cost of restoring the output word sequence to the original input sequence. A single summary statistic is required to permit graphical comparison of system performance, and this should have an intuitive interpretation.

For Isolated Word Recognition (IWR) WER is defined as

$$\text{WER(IWS)} = \frac{S}{N = H + S} = 1\frac{-H}{N} \quad (3)$$

Let *H, S, D* and *I* denote the total number of word hits, substitutions, deletions and insertions and *N* denote the total number of input words, output words, and matched I/O word pairs.

WER in CSR(Connected Speech Recognition) is then defined as the ratio of the number of errors to the number of words input.

$$\text{WER(CSR)} = \frac{S + D + I}{N_1 = H + S + D} \quad (4)$$

Speed is measured in terms of Real Time Factor (RTF) which is given in equation :

$$RTF = P/I \quad (5)$$

Where P is the time that will be taken to process input of I duration.

Previous system has a problem of accuracy and speed but now proposed system is trying to overcome this problem that helps to increase the quality of system.

7. Proposed Systems

There are basically four steps to convert the speech into text that are follows:

A Audio to phonemes

B Phonemes to words

C Words to phrases

D Raw transcribed text to formatted text

Initially the input signal is transformed from analog to digital form and then it is divided into frames each having their individual frequency. The spectral features extracted from the frames help in identifying the phones of the speech sample which are nothing but sounds that distinguish two words. MFCC is applied to the phones which identify unique discrete acoustic phone of each individual input speech sample. Then HMM is applied to each phone which identifies the likelihood of occurrence of a word W within the given acoustic observation P (W/A).

The probability of occurrence of a word

P(W/A), is given as follows (using Baye's rule format)

$$P(W/A) = \frac{P(A/W)\,P(W)}{P(A)} \quad (6)$$

Since P (A) is independent of W, the MAP decoding rule

$$W = \text{argmax}_w\, P(A/W)\, P(W) \quad (7)$$

1. P(A/W) is called the acoustic model which will check the active vocabulary and vocabulary for database through which the probability of pitch level is maintained.

2. P (W) is called the language model. It describes the probability associated with a postulated sequence of words. These type of models give the semantic and syntactic values of the task.

N-Gram dataset is a database that contains the text feature vector with the approach of set of words to solve my computational problems. With the help of N-Gram dataset, for the proposed system vocabulary dictionary and sentences are available which act as a huge repository for both training as well as testing phase of the proposed system. An algorithm called sound X is then created. This algorithm finds the best possible match from all the possible patterns returned by N-Gram dataset.

The proposed error correction method combines three algorithms: The error detection algorithm,the candidate corrections generation algorithm and the context-sensitive error correction algorithm. This is done by sound X Algorithm into three modules.

```
//Error Detection Algorithm
FunctionErrorDetection (A)
{
// split the ASR text on space and return word tokens
W<-Split(A,“”)
for(i<-0 to i < N) // detect all word tokens
{
// search for W[i] in Microsoft N-Gram dataset
R<-Search(MicrosoftDataset , W[i])
if(R == true) // mean W[i] was found in Microsoft dataset(i.e. correctly spelled)
i<- i+1 // go to the next word tokenW[i+1]
else // W[i] is misspelled and thus a correction is required
// go to the candidate corrections generation algorithm
GenerateCandidates(W[i])
}
//The Candidate Corrections Generation Algorithm
Function GenerateCandidates (word)
{
// create 2-gram character sequences and put them
a<-Split2Grams(word)
for(i<-0 to i < N) // for all 2-gram sequences
{
// look for unigrams having a[i] as substring, (i.e.
//unigrams sharing 2-gram sequence with the error word
L[i] <-Substring(MicrosoftDataset, a[i])
i<- i+1
}
// select the top 8 unigrams sharing 2-gram character sequences with the error word candidates
<-commonUnigrams(L)
// go to the error correction algorithm
Error Correction(candidates)
}
//The Error Correcting Algorithm
Function Error Correction (candidates)
{
for(i<-0 to i < N) // process all candidate corrections
{
// concatenate together the ith candidate with the four preceding words
// A is a global array containing the original ASR output text
L <-Concatenate(A[j<-4] , A[j<-3] , A[j<-2] , A[j<-1] , candidates[i] )
// find L in Microsoft N-gram dataset and returns its frequency
frequency[i] <- Search(MicrosoftDataset , L)
i <- i+1
}
p<-MaxFrequency(frequency)
```

// return the index p of the candidate whose L has the highest frequency

// return the correction for the ASR error

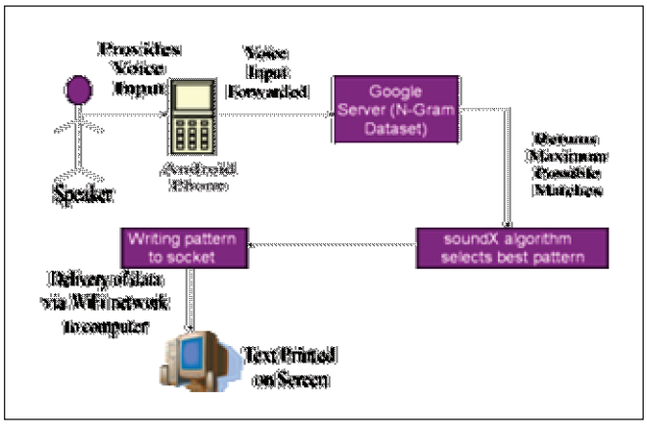
RETURN candidates[p]

}

8. Experimental Setup

This figure describes the working of proposed work.

Figure 3. Working Principle



The objective of proposed work is to print the text on screen through mobile phones not through common means like keyboard, mouse but through input speech using android mobile phone. In order to perform this work, the input speech feature is extracted from acoustic and language model. Acoustic model contains the active vocabulary to extract the pitch level and maintain the database through HMM and MFCC. Language model describes the pronunciation of speech without disfluency. The speaker provides voice input from android phone and send it to the Google server by using feature extraction technique of HMM and MFCC. Google server database uses the n- gram dataset. Google server finds the best possible match with the help of n- gram dataset. The sound X algorithm selects the best match. The selected match gets printed on the screen.

The Soundx algorithm filters the pattern by:

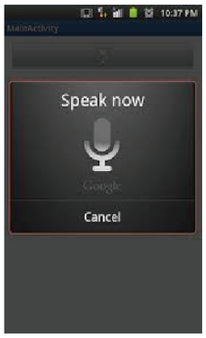
```
private static char[ ] makeRandomChars(Random r, final int size) {
  char[] data = new char[size];
  for (int i = 0; i < data.length; i++) {
   data[i] = (char) (r.nextInt(255 - 32) + 32);
  }
  return data;
  }
```

This is done by making the socket programming and use the WIFI network to establish the connection between the computer and android mobile phone.

9. Results

The following interfaces depict how the system is working. Initially the user opens app called VoiceExample in his android phone and presses the "Speak Click" button. User then provides some voice input. For example, here the user provides an input as "Welcome to working application". The speech input finds the best possible match through HMM and MFCC and then selects the best match through soundX algorithm and gets printed on screen i.e on notepad.

Figure 4. Interface of Before Processing the Application

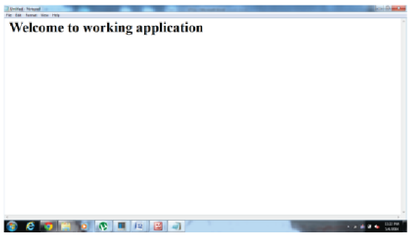


Then it goes to process to find the match of voice input.

Figure 5. To Process the Request

The Output gets printed on screen and illustrated as:

Figure 6. After Executing the Request Printed Output



As discussed in the paper above about the problem statement of performance about WER and RTF on speech recognizer. For this system RTF is calculated as:

$$RTF = 4.6/5 = 92\% \quad (8)$$

where 4.6 seconds P processing time is taken for an input of duration I of 5 seconds.

On the basis of the interfaces shown above, WER is calculated for the input speech: "Welcome to working application" whose output that is returned by the system is: "Welcome wapking application". As it is evident from the interfaces, for N = 5 total words spoken by speaker; S = 1 substitutions are made as 'working' is replaced by 'wapking'; D = 1 as 'the' is deleted from the N-Gram dataset returned output. Therefore, WER is:

$$WER = (1 + 0 + 1)/5 = 40\% \quad (9)$$

10. Analysis of Existing Systems

The table 1 tells the performance of different existing system about the accuracy on the basis of feature extraction technique by choosing the LPC (Linear Predictive Coding), MFCC (Mel Frequency Cepstral Coefficient) and PLP (Perceptual Linear Prediction) and which recognition technique they choose out of HMM (Hidden Markov Model), GA (Genetic Algorithm) and VQ (Vector Quantization).

11. Conclusion and Future Work

Speaker independent continuous speech recognition systems with large vocabulary are in-demand which can be fulfilled by using the feature extraction technique MFCC with the recognition technique HMM which help in creating extremely powerful systems that offer good speech recognition results. More than 60 years is the legacy of speech recognition system. The ASR systems prove to be useful not only for blind people but also let able people to do some other work as their hands and eyes are free to indulge in other activities.

For future work, the future systems must be trained to withstand a URL typed in browser's address bar. The future system will attaching files to e-mails. These systems will deploy entire system for mobility and increase duration of input. The additional feature could be to control entire computer's function via speech input which may range from typing something to giving a command to it like opening a drive or copying something or operating internet; just anything and everything.

Table1 : Comparison of Existing Systems

Research Work Name	*Feature Extraction Technique*	*Recognition Technique*	*Accuracy*
Alaigal-A Tamil Speech Recognition [17]	PLP	HMM	70% - 80%
Arabic Speech Recognition Using Hidden Markov Model Toolkit (HTK) [19]	MFCC	HMM	97.99%
Automatic Speech Recognition:Human Computer Interface for Kinyarwanda Language [33]	MFCC	HMM	92%
Automatic Speech Recognition for Bangla Digits [26]	MFCC	HMM	More than 95% for digits (0-5) and less than 90% for digits (6-9)
Continuous Speech Recognition System for Tamil Using Monophone-based Hidden Markov Model [44]	MFCC	HMM	92% accuracy in word level and 81% accuracy in sentence level
English Digits Speech Recognition System Based on Hidden Markov Models [18]	MFCC	HMM	56.25% - 72.5%

Hindi Speech Recognition System Using HTK [29]	MFCC	HMM	Word-accuracy and word-error rate of the system are 94.63% and 5.37% respectively
Human Computer Interaction Using Isolated Words Speech Recognition Technology [39]	MFCC	VQ	88%
Segment-Based Stochastic Modelings for Speech Recognition [52]	LPC	HMM + VQ	62% - 96%

References

1. Morris, A. C., Maier, V. & Green, P. (1999). *From WER and RIL to MER and WIL: Improved evaluation measures for connected speech recognition*. Institute of Phonetics Saarland University, Germany.
2. Lee, S., Kang, S. & Hanseok, K. O., Jongseong, Y. & Minseok, K. (2013). *Dialogue Enabling Speech-to-Text User Assistive Agent with Auditory Perceptual Beam forming for Hearing-Impaired*. International Conference on Consumer Electronics (ICCE).
3. Minematsu, N., Saito, D. & Hirose, K. (2008). *Experimental Study of Structure to Speech Conversion*. 9th International Conference on Signal Processing (pp. 651-654).
4. Gyo¨rgy, S. A.´, Kos Ma´te´, T. & Kla´ra, V. (1997). *Automatic Speech to Text Transformation of Spontaneous Job Interviews on the HuComTech Database*. Budapest University of Technology and Economics.
5. Jaiswal, P. K. & Mishra, P. K. (2012). A review of speech pattern recognition survey. *International Journal of Computer Science and Technology*, 3(1), 709-713.
6. Bansal, A., Kant, K. & Chauhan, K. (2013). A review of speech recognition system. *COMPTECH: An International Journal of Computer Sciences*, January, 3(6), 709-713.
7. Acero, A. (2000). *An Overview of Text-to-Speech Synthesis*. Speech Technology Group Microsoft Corporation Redmond. Proceedings of 2000 IEEE Workshop on Speech Coding.
8. Lamel, L., Gauvain, J. L., Le, V. B., Oparin, I. & Meng, S. (2011). *Improved Models for Mandarin Speech-to-text Transcription*. IEEE International Conference on Acoustics, Speech and Signal Processing (pp. 4660-4663).
9. Adell, J., Ag¨uero, P. D. & Bonafonte, A. (2006). *Database Pruning for Unsupervised Building of Text-To-speech Voices*. Department of Signal Theory and Communications. TALP Research Center.
10. Tuerk, C., Monaco, P. & Robinson, T. (1991). *The Development of A Connectionist Multiple-voice Text-to-speech System*. Cambridge University Engineering Department.
11. Amarasekara, M. S., Bandara, K. M. N. S., Yithana, B. Y. A. I., Silva, O. H. & Jayakody. (2013). *Real Time Interactive Voice Communication.* The 8th International Conference on Computer Science & Education (ICCSE 2013) April, (pp. 26-28).
12. McCowan, I., Moore, D., Dines, J., Gatica-Perez, J., Flynn, M., Wellner, P. & Bourlard, H. (2005). On The Use of Information Retrieval Measures for Speech Recognition Evaluation. IDIAP Research Report.
13. Alter, R. (1968). *Utilization of Contextual Constraints in Automatic Speech Recognition*. IEEE Transactions on Audio and Electroacoustics, March, 16(1), 6-11.
14. Diehl, F., Gales, M. J. F., Tomalin, M. & Woodland, P. C. (2008). *Phonetic Pronunciations for Arabic Speech-to-text Systems*. Engineering Department, Cambridge University, Trumpington St., Cambridge.
15. F¨urnkranz, J. (1998). A Study Using n-gram Features for Text Categorization. Austrian Research Institute for Artificial Intelligence. Technical Report OEFAI-TR-98-30.
16. Mukhopadhayay, A., Chakraborty, S., Choudhary, M., Lahiri, A., Dey, S. & Basu, A. (2006). *Shruti: An Embedded Text-to Speech System for Indian Languages*. IEEE Proceedings on Software Engineering, April, 153(2), 75-79.
17. Henry, A. P. & Devaraj, C. G. (2004). *Alaigal- A Tamil Speech Recognition*. Tamil Internet Singapore.
18. Abushariah, A. A. M., Gunawan, T. S., Abushariah, M. A. M. & Khalifa, O. O. (2010). *English Digits Speech Recognition System Based on Hidden Markov Models*. International Conference on Computer & Communication Engineering (pp. 978).
19. Al-Qatab, B. A. Q. & Ainon, R. N. (2010). *Arabic Speech Recognition Using Hidden Markov Model Toolkit (HTK)*.

20. Singh, B., Kapur, N. & Kaur, P. (2012). Speech recognition with hidden Markov model: A review. *International Journal of Advanced Research in Computer Science and Software Engineering*, 75(2), 8-15.
21. Heerden, C. V., Barnard, E., Feld, M. & Miller, C. (2010). *Combining Regression & Classification Methods for Improving Automatic Speaker Age Recognition*. Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing.
22. Corfield, C. (2012). *Demystifying Speech Recognition*. nVoq White Paper.
23. Yeo, C. Y., Al-Haddad, S. A. R. & Ng, N. G. (2011). *Animal Voice Recognition for Identification (ID) Detection System*. IEEE 7th International Colloquium on Signal Processing & it's Applications.
24. Kaur, J., Nidhi. & Kaur, R. (2012). Issues involved in speech to text conversion. *International Journal of Computational Engineering Research*, March-April, 2(2), 512-515.
25. Sharma, R. & Wason, G. (2012). *Speech recognition & synthesis tool: Assistive technology for physically disabled persons. International Journal of Computer Science & Telecommunications*, April, 3(4), 86-91.
26. Muhammad, G., Alotaibi, Y. A. & Huda, M. N. (2009). *Automatic Speech Recognition for Bangla Digits*.
27. Zhiyan, H., Shxian, L. & Jian, W. (2012). *Speech Emotion Recognition System based on Integrating Feature and Improved HMM*. The 2nd International Conference on Computer Application and System Modeling.
28. Sarfaraz, H., Hussain, S., Bokhari, R., Raza, A. A., Ullah, I., Sarfaraz, Z., Pervez, S., Mustafa, A., Javed, I. & Praveen, R. (2010). *Large Vocabulary Continuous Speech Recognition for Urdu*. International Conference on the Frontiers of Information Technologies (December, pp. 21-23), Islamabad, Pakistan.
29. Kumar, K. & Aggarwal, R. K. (2011). *Hindi Speech Recognition System using HTK*. International Journal of Computing and Business Research, 2(2).
30. Myers, L. (2004). *An Exploration of Voice Biometrics*. SANS Institute Infosec.
31. Mangu, L. & Padmanabhan, M. (2001). *Error Corrective Mechanisms for Speech Recognition*. International Conference on Acoustics, Speech and Signal Processing.
32. Chandrashekhar, M., & Ponnavaikko, M. (2008). Tamil Speech Recognition: A Complete Model. *Electronic Journal Technical Acoustics* (pp. 1-15)
33. Jackson, M. (2005). *Automatic Speech Recognition: Human Computer Interface for Kinyarwanda Language*. Master Thesis, Faculty of Computing & Information Technology, Makerere University.
34. Akram, M. U., & Arif, M. (2004). *Design of an Urdu Speech Recognizer Based Upon Acoustic Phonetic Modeling Approach*. In proceedings of 8th International Conference on Multitopic INMIC.
35. Bohac, M. (2012). *Performance Comparison of Several Techniques to Detect Words in Audio Streams and Audio Scene*. IEEE 54th International Symposium ELMAR-2012.
36. Pleva, M., Ondas, S., Juhar, J., Cizmar, A., Papaj, J., Dobos, L. (2011). *Speech & Mobile Technologies for Cognitive Communication & Information Systems*. IEEE Proceedings of 2nd International Conference on Cognitive InfoCommunications (CogInfoCom).
37. Feld, M., Barnard, E., Heerden, C. V., & Muller, C. (2009). *Multilingual Speaker Age Recognition: Recognition Analyses on the Lwazi Corpus*. IEEE Workshop on Automatic Speech Recognition & University (pp. 534-539).
38. Grimm, M., Kroschel, K., & Narayanana, S. (2008). *The Vera Am Mittag German Audio-Visual Emotional Speech Database*. In proceedings of the Multimedia & Expo.
39. *Abushariah*, M. A. M., *Zainuddin*, R. N. A. R., Abmhara, M., *Khalifa*, O. O. (2007). *Human Computer Interaction Using Isolated Words Speech Recognition Technology*. International Conference on Intelligent & Advanced Systems.
40. Ilyas, M. Z., Samad, S. A., Hussain, A., Ishak, K. A. (2007). *Speaker Verification using Vector Quantization and Hidden Markov Model*. The 5th Student Conference on Research & Development.
41. Morgan, N. (2012). *Deep and Wide: Multiple Layers in Automatic Speech Recognition*. IEEE Transactions on Audio, Speech & Language Processing, January, 20(1) 7-13.
42. Soluade, O. A. (2009). *A Comparative Analysis of Speech Recognition Platforms*. Communications of the IIMA.
43. Chen, O. T. C., Gu, J. J., Lu, P. T. & Ke, J. Y. (2012). *Emotion Inspired Age and Gender Recognition Systems*.
44. Radha, V., Vimala, C., & Krishnaveni, M. (2012).

Continuous Speech Recognition System for Tamil Language using Monophone-based Hidden Markov Model. Proceedings of the Second International Conference on Computational Science, Engineering and Information Technology (pp. 227-231).

45. Bettelheim, R., & Steele, D. (2010). *Speech and Command Recognition*. FreeScale White Paper.
46. Basu, S., Neti, C., Rajput, N., Senior, A., Subramaniam, L., & Verma, A. (1999). *Audio-Visual Large Vocabulary Continuous Speech Recognition in Broadcast Domain*. IEEE 3rd Workshop on Multimedia Signal Processing (pp. 475-481).
47. Mandal, S., Das, B., & Mitra, P. (2010). *Shruti-II: A Vernacular Speech Recognition System in Bengali and an Application for Visually Impaired Community*. IEEE Student's Technology Symposium (pp. 229-233).
48. Primorac, S., & Russo, M. (2012). *Android Application for Sending SMS messages with Speech Recognition Interface*.
49. Gaikward, S. K., Gawali, B. W., & Yannawar, P. (2010). *A Review on Speech Recognition Technique*. *International Journal of Computer Applications*, 10(3), 16-24.
50. *Speech Recognition Technology Choices*. (2010). A Vocollect White Paper.
51. Tiwari, V. (2010). MFCC and it's applications in speaker recognition. *International Journal on Emerging Technique*, 1(1), 19-22.
52. Vimala C., & Radha, V. (2012). A review on speech recognition challenges and approaches. *World of Computer Science and Information Technology Journal,* 2(1), 1-7.
53. Xiaofeng, W., & Nakatsu, R. (2010). *Vision-aided Speech Recognition System for a Small Four-Legged Robot*. International Conference on Audio Language & Image Processing (1073-1078).
54. Jian, Y., & Jin, J. (2012). *An Interactive Interface between Human & Computer based on Pattern & Speech Recognition*. International Conference on Systems & Informatics.
55. Basil, Y. & Semaan, P. (2012). ASR context-sensitive error correction based on microsoft n-gram dataset. *Journal of Computing*, 4(1), 34-42.
56. Hachkar, Z., Mounir, B., Farchi, A., Abbadi, J. E. (2011). Comparison of MFCC and PLP parameterization. *Canadian Journal on Artificial Intelligence, Machine Learning & Pattern Recognition,* April, 2(3), 41-55.
57. Lishuang, Z., & Zhiyan, H. (2010). *Speech Recognition System Based on Integrating Feature and HMM*.
58. Ng, T., Zhang, B., Nguyen†, K., & Nguyen, L. (2008). *Progress in the Bbn 2007 Mandarin Speech to Text System*. BBN Technologies, 10 Moulton Street, Cambridge.
59. Rajalakshmi, R., & Revathy, A. (2012). *Comparison of MFCC and PLP in Speaker Identification using GMM*. International Conference on Computing and Control Engineering.