Journal of Signal and Information Processing, **2012, 3, 394-401**
http://dx.doi.org/10.4236/jsip.2012.33052 Published Online August 2012 (http://www.SciRP.org/journal/jsip)

Development of Application Specific Continuous Speech Recognition System in Hindi

Gaurav, Devanesamoni Shakina Deiv, Gopal Krishna Sharma, Mahua Bhattacharya*

Indian Institute of Information Technology & Management, Gwalior, India.
Email: *bmahua@hotmail.com

Received April 1st, 2012; revised May 3rd, 2012; accepted May 13th, 2012

ABSTRACT

Application specific voice interfaces in local languages will go a long way in reaching the benefits of technology to rural India. A continuous speech recognition system in Hindi tailored to aid teaching Geometry in Primary schools is the goal of the work. This paper presents the preliminary work done towards that end. We have used the Mel Frequency Cepstral Coefficients as speech feature parameters and Hidden Markov Modeling to model the acoustic features. Hidden Markov Modeling Tool Kit −3.4 was used both for feature extraction and model generation. The Julius recognizer which is language independent was used for decoding. A speaker independent system is implemented and results are presented.

Keywords: Automatic Speech Recognition; Mel Frequency Cepstral Coefficients; Hidden Markov Modeling

1. Introduction

To make Information Technology (IT) relevant to rural India, voice access to a variety of computer based services is imperative. Although many speech interfaces are already available, the need is for speech interfaces in local Indian languages. Application specific Hindi speech recognition systems are required to make computer aided teaching, a reality in rural schools. This paper presents the preliminary work done to demonstrate the relevance of a Hindi Continuous Speech Recognition System in primary education.

Automatic speech recognition has progressed tremendously in the last two decades. There are several commercial Automatic Speech Recognition (ASR) systems developed, the most popular among them are Dragon Naturally Speaking, IBM Via voice and Microsoft SAPI. Efforts are on to develop speech recognition systems in different Indian Languages. An isolated word Hindi ASR for small vocabulary is developed and evaluated in [1]. An effort to increase the recognition accuracy of Hindi ASR by online speaker adaptation has been reported in [2]. It is demonstrated that Maximum Likelihood Linear Regression (MLLR) transform based adaptation transforms the acoustic models in such a way that the difference between test and training conditions is reduced, resulting in better performance.

A general approach to identifying feature vectors that effectively distinguish gender of a speaker from Hindi vowel phoneme utterances has been presented in [3,4]. Centre for Development of Advanced computing has developed a domain specific speaker independent continuous speech recognition system for Hindi using Julius recognition engine [5]. They also have built a Hindi ASR for travel domain [6] giving encouraging recognition accuracy.

State likelihood evaluation in Hidden Markov model (HMM) using mixture of Gaussians is one problem that needs to be solved. A novel method using Gaussian Mixture Model (GMM) for statistical pattern classification is suggested to reduce computational load [7]. Development of speech interfaces in Hindi for IT based services is a work in progress [8]. Efforts to compensate for different accents in Hindi are also explored in [9]. Apart from Hindi ASR, speech recognition systems are being developed in other languages like Arabic, Malayalam, Tamil, Bengali, Telugu, etc. [10-14].

IBM Research Laboratory of India has developed a Hindi Speech Recognition system which has been trained on 40 hours of audio data and has a trigram language model that is trained with 3 million words [15]. Efforts are on to develop large speech databases in various Indian Languages for Large Vocabulary Speech Recognition Systems [16]. SRI Language Model (SRILM) extensible toolkit is discussed in [17] which can be used for developing Language model. This toolkit has been used in developing language model for large vocabulary sys-

*Corresponding author.

tems in Hindi.

Hidden Markov Model provides an elegant statistical framework for modeling speech patterns and is the most widely used technique [18,19]. Recently the hybrid HMM and Artificial Neural Network (ANN) framework is also used in an effort to overcome the challenges posed by speech variability due to physiological differences, style variability due to co-articulation effects, varying accents, emotional states, context variability etc [20].

Another method to handle the problem of changes in the acoustic environment or speaker specific voice characteristics is by adapting the statistical models of a speech recognizer and speaker tracking. Combining speaker adaptation and speaker tracking may be advantageous, because it allows a system to adapt to more than one user at the same time. Authors in [21] have extended a standard speech recognizer by combining speaker specific speech decoding with speaker identification in an efficient manner. Approximately 20% relative error rate reduction and about 94.6% identification rate are reported.

The system presented here is an application specific Continuous Speech Recognizer in Hindi. It is restricted to the task of computer-aided teaching of Geometry at primary school level. The paper is organized as follows. Section 2 describes the architecture of the speech recognition system with the function of each module. Section 3 explains the training methodology of developing the proposed Hindi CSR. Section 4 details the testing of the system. The results are discussed in Section 5. Section 6 concludes with future direction of the work.

2. Automatic Speech Recognition System

Speech recognition is the process of converting an acoustic signal, captured by a microphone or a telephone into a set of words. The recognized words can be the final result for applications such as commands and control, data entry, and document preparation. They can also serve as the input to further linguistic processing in order to achieve speech understanding. **Figure 1** shows the block diagram of a state of the art automatic speech recognition system.

Speech signal is analog. In the first place analog electrical signals are converted to digital signals. This is done in two steps, sampling and quantization. So a typical representation of a speech signal is a stream of 8-bit numbers at the rate of 10,000 numbers per second. Once the signal conversion is complete, background noise is filtered to keep signal to noise ratio high. The signal is pre-emphasized and then speech parameters are extracted.

2.1. Feature Extraction

Mel-Frequency Cepstral Coefficients (MFCCs) are widely used features for automatic speech recognition systems to transform the speech waveform into a sequence of discrete acoustic vectors.

The MFCC technique makes use of two types of filter, namely, linearly spaced filters and logarithmically spaced filters. The Mel frequency scale has linear frequency spacing below 1000 Hz and a logarithmic spacing above 1000 Hz. In the sound processing, the Mel-frequency cepstrum 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-frequency scale.

The procedure by which the Mel-frequency cepstral coefficients are obtained consists of the following steps. **Figure 2** depicts the procedure of extracting MFCC feature vectors from speech.

The signal is passed through a filter which emphasizes higher frequencies. This process will increase the energy of the signal at higher frequency.

The Pre-emphasis of the speech signal is realized with this simple FIR filter

$$H(z) = 1 - az^{-1} \tag{1}$$

where a is from interval [0.9, 1].

The digitized speech is segmented into frames with a length within the range of 10 to 40 ms. The segment of waveform used to determine each parameter vector is usually referred to as a window.

The Hamming window which is used for the purpose is defined by the equation

$$w(n)0.54 - 0.46\cos\left[2\pi n/(N-1)\right] \tag{2}$$

where, $0 \le n \le N-1$

N = number of samples in each frame.

Let $Y(n)$ = Output signal and $X(n)$ = input signal

The result of windowing the signal is

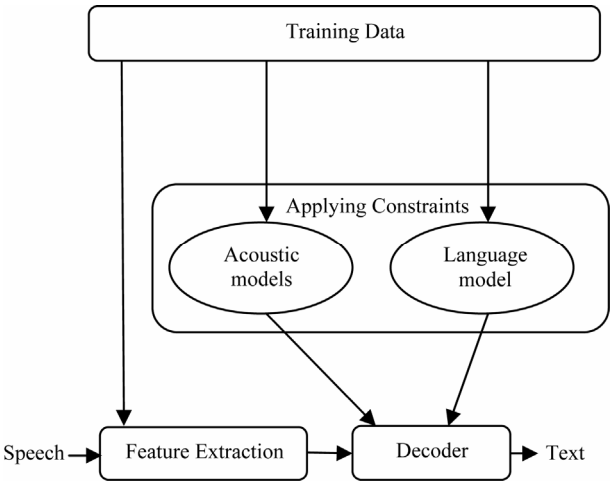

Figure 1. Automatic speech recognition system.

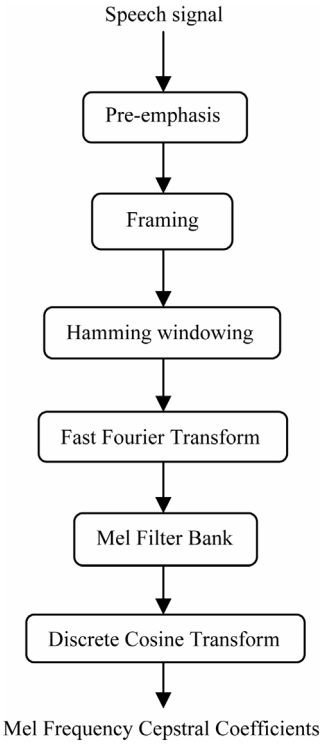

Figure 2. Flow chart of MFCC feature extraction.

$$Y(n) = X(n)W(n) \tag{3}$$

Next, the Fast Fourier transform (FFT) is used to convert each frame of N samples from time domain into frequency domain. Thus the components of the magnitude spectrum of the analyzed signal are calculated.

$$Y(\omega) = \mathrm{FFT}\left[h(t) * x(t)\right] = H(\omega)X(\omega) \tag{4}$$

The most important step in this signal processing is Mel-frequency transformation. Compensation for non-linear perception of frequency is implemented by the bank of triangular band filters with the linear distribution of frequencies along the so called Mel-frequency range. Linear deployment of filters to Mel-frequency axis results in a non-linear distribution for the standard frequency axis in hertz. Definition of the Mel-frequency range is described by the following equation.

$$f_{mel} = 2595\log_{10}(1 + f/100)\,Hz \tag{5}$$

where f is frequency in linear range and f_{mel} the corresponding frequency in nonlinear Mel-frequency range.

The Mel spectrum coefficients and their logarithm are real numbers. Hence they can be converted to the time domain using the discrete cosine transform (DCT). The result is the Mel Frequency Cepstral Coefficients. The cepstral representation of the speech spectrum provides a good representation of the local spectral properties of the signal for the given frame analysis.

$$c_n = \sqrt{2/K}\sum_{j=1}^{N}\left(\log_{mj}\right)\cos\left(\pi n(j-0.5)/K\right) \tag{6}$$

n = number of Mel-frequency cepstral coefficients

K = number of Mel-frequency band filters (filter bank channels) in the bank of filters.

2.2. The Acoustic Model

In a statistical framework for speech recognition, the problem is to find the most likely word sequence, which can be described by the equation

$$\hat{W} = \arg_w \max P(W/X) \tag{7}$$

Applying the Bayes' equation, we get

$$\hat{W} = \arg_w \max P(W/X)P(W) \tag{8}$$

The term $P(X/W)$ in the above equation can be realized by the Acoustic model. An acoustic model is a file that contains a statistical representation of each distinct sound that makes up a spoken word. It contains the sounds for each word found in the Language model.

The speech recognition system implemented here uses Hidden Markov Models (HMM) for representing speech sounds. A HMM is a stochastic model. A HMM consists of a number of states, each of which is associated with a probability density function. The model parameters are the set of probability density functions, and a transition matrix that contains the probability of transitions between states.

HMM-based recognition algorithms are classified into two types, namely, phoneme level model and word-level model. The word-level HMM has excellent performance at isolated word tasks and is capable of representing speech transitions between phonemes. However, each distinct word has to be represented by a separate model which leads to extremely high computation cost (which is proportional to the number of HMM models). The phoneme model on the other hand can help reproduce a word as a sequence of phonemes. Hence new words can be added to the dictionary without necessitating additional models. Hence phoneme model is considered more suitable in applications with large sized vocabularies and where addition of word is a essential possibility.

The phoneme model is used here. The MFCC features extracted from speech and the associated transcriptions are used to estimate the parameters of HMM based acoustic models that represent phonemes. The iterative process of estimating and re-estimating the parameters to achieve a reasonable representation of the speech unit is called ASR system training. The training procedure involves the use of forward-backward algorithm.

2.3. The Language Model

The term $P(W)$ in Equation (2) represents the *a priori*

probability of a word sequence based on syntax, semantics and pragmatics of the language to be recognized. It can be realized by the Language Model which contains a list of words and their probability of occurrence in a given sequence, and is independent of the acoustic signal. The probability of a word sequence is given below.

$$p(w_1, w_2, w_3, w_4, w_5, \cdots, w_n) = p(W) \quad (9)$$

By Chain rule the probability of nth word is:

$$p(w_1^n) = p(w_1)\, p(w_2|w_1)\, p(w_2|w_1^2) \cdots p(w_n|w_1^{n-1}) \quad (10)$$

$$p(w_1^n) = \prod_{k=1}^{n} p(w_k|w_1^{k-1}) \quad (11)$$

Language Model or Grammar essentially defines constraints on what the Speech Recognition Engine can expect as input.

2.4. The Recognizer

Recognizer is a Software program that takes the sounds spoken by a user and searches the Acoustic Model for the equivalent sounds. When a match is made, the Decoder determines the phoneme corresponding to the sound. It keeps track of the matching phonemes until it reaches a pause in the user's speech. It then searches the Language Model or Grammar file for the equivalent series of phonemes. If a match is made it returns the text of the corresponding word or phrase to the calling program.

3. The Training Methodology

The two major stages involved in the process of Speech Recognition are the training of the ASR and the Testing. The training phase involves the following steps.

3.1. The Database

The text corpus consists of chosen application specific sentences, pertaining to teaching Geometry to children. Forty three distinct Hindi sentences about shape geometry using 29 distinct Hindi phonemes were designed as the text corpus. These sentences were spoken in a continuous fashion and recorded using good quality microphones under office noise conditions.

The Wave-surfer software was used for recording. The training corpus contains 1806 utterances spoken by 12 females and 18 males. All the speakers are natives of the Hindi heart-land of India, educated and in the age group of 18 to 30.

3.2. Phone Set

Phoneme is the basic unit of sound in any language. Hindi belongs to the Indo Aryan family of languages and is written in the Devanagari script. There are 11 vowels and 35 consonants in standard Hindi. In addition, five Nukta consonants are also adopted from Farsi/Arabic sounds. The phone set that is used here to develop the application specific speech recognition system for Hindi language uses only 29 of the 60 used in large vocabulary systems.

3.3. Lexicon

The pronunciation dictionary (lexicon) contains all the distinct words in the corpus and its corresponding pronunciation given as a string of phonemes. Some sample entries are given in **Table 1**. The pronunciation dictionary is case insensitive. This dictionary includes entries for the beginning-of-sentence and the end-of-sentence tokens and respectively as well as the silence.

3.4. Transcription

The transcription file contains the sentences or utterances of the spoken text and the corresponding audio files in the following format. Each word in the transcription file is present in the pronunciation lexicon.

3.5. Parameterization of Speech Data

The digitized speech signal is subjected to first order pre-emphasis applied using a coefficient of 0.97. The signal is then segmented into frames and hamming windowed. The HMM Tool Kit (HTK) [22] was used to parameterize the raw speech waveforms into sequences feature vectors.

Table 1. Pronunciation lexicon.

आयत	aa y ax t sp
बैगनी	b ae g n iy sp
बनाओ	b ax n aa ow sp
चतुर्भुज	ch ax t uh r b hh uh jh sp
एक	ey k sp
हरा	hh ax r aa sp
काला	k aa l aasp
लाल	l aa l sp
नारंगी	n aa r ax ng iy sp
नीला	n iy l aa sp
पीला	p iy l aa sp
सफ़ेद	s ax f eh ey dh sp
समान्तर	s ax m aa n t ax r sp
सम्बाहू	s ax m b aa hh uh sp

Mel Frequency Cepstral Coefficients (MFCCs) are derived from FFT-based log spectra. Coding was performed using the tool HCopy configured to automatically convert its input into MFCC vectors. A configuration file specifies all of the conversion parameters [22]. A typical configuration file is seen in **Figure 3**.

The target parameters are to be MFCC using C_0 as the energy component. The standard 39 dimension MFCC, delta and acceleration feature vector is computed for the 16 kHz sampled signals at 10 ms intervals (100 ns). Mel scaled 26 filter banks spanning the 8 kHz frequency range are used for computation of MFCCs.

The output was saved in compressed format, and a crc checksum added. The 39-demensional feature vector consists of 13 Mel Scale Cepstral Coefficients and their first and second derivates. A sample MFCC file is shown below in **Figure 4**.

3.6. Acoustic Model Generation

The speech recognition system implemented here employs Hidden Markov Model (HMM) for representing speech sounds. A HMM consists of a number of states, each of which is associated with a probability density function. The parameters of a HMM comprises of the parameters of the set of probability density functions, and a transition matrix that contains the probability of transition between states.

The MFCC feature vectors extracted from speech signals and their associated transcriptions are used to estimate the parameters of HMMs. This process is called ASR system training. HMM Tool Kit, HTK-3.4 was used for training models over 29 context-dependent Hindi phonemes used in the chosen application. The basic acoustic units are context dependent phonemes, that is, tri-phones modeled by left-to-right, 5-state, HMMs.

The output probability distributions of states were represented by Gaussian mixture densities. For every state of every phoneme 256 global Gaussian density functions were used to generate Gaussian mixtures.

Prototype models are built using the flat start approach. With the exception of the transition probabilities, all of the HMM parameters given in the prototype definition are ignored. The purpose of the prototype definition is only to specify the overall characteristics and topology of the HMM. The actual parameters will be computed later.

```
# Coding parameters
SOURCEFORMAT = WAV
TARGETKIND = MFCC_0_D_A
TARGETRATE = 100000.0
SAVECOMPRESSED = T
SAVEWITHCRC = T
WINDOWSIZE = 250000.0
USEHAMMING = T
PREEMCOEF = 0.97
NUMCHANS = 26
CEPLIFTER = 22
NUMCEPS = 12
ENORMALISE = F
```

Figure 3. A typical configuration file.

```
---------- Observation Structure
x:      MFCC-1  MFCC-2  MFCC-3  MFCC-4  MFCC-5  MFCC-6  MFCC-7  MFCC-8  MFCC-9 MFCC-10 MFCC-11 MFCC-12      C0   Del-1   Del-2   Del-3   Del-4   Del-5   Del-6
  Del-7   Del-8   Del-9  Del-10  Del-11  Del-12   DelC0   Acc-1   Acc-2   Acc-3   Acc-4   Acc-5   Acc-6   Acc-7   Acc-8   Acc-9  Acc-10  Acc-11
        Acc-12   AccC0
---------- Samples: 0->250
0:     -22.999  -2.981  -4.676  -1.275  -2.580   1.256   4.270   0.432  -2.956  -5.046  -3.683  -0.171  21.748   2.614  -0.068   0.160  -0.005   0.631   0.100
  -0.880  -2.079   0.097   0.572   0.974   0.336   3.511  -0.096  -0.136   0.064  -0.145   0.184  -0.431  -0.106   0.468   0.168   0.028  -0.099
        -0.064   0.325
1:     -16.113  -1.304  -3.702   0.283   0.586   3.377  -0.922  -7.323  -4.431  -0.708   3.515   1.653  27.271   2.858  -0.124   0.588  -0.029   1.311  -0.884
  -1.482  -1.775   0.767   0.751   0.944   0.712   4.693  -0.564  -0.198  -0.048  -0.378  -0.111  -0.538   0.155   1.059   0.251   0.007  -0.147
        -0.179  -0.048
2:     -13.372  -4.158  -4.363  -2.081  -1.007   0.695   2.464  -6.085  -1.735  -4.352  -2.414   0.597  36.543   2.012  -0.722   0.267  -0.716   1.214  -1.563
  -1.111   0.109   0.603   0.625   0.495  -0.171   4.546  -0.878  -0.048  -0.054  -0.317  -0.502  -0.151   0.345   1.000   0.306   0.022  -0.150
        -0.467  -0.715
3:     -13.522  -3.010  -1.893  -1.017   3.186  -2.882  -2.239  -5.184   0.267  -1.638   0.402   3.006  37.818   0.095  -0.730  -0.134  -1.539  -0.217  -1.759
   0.010   2.122   1.100   0.582   0.480  -0.304   2.753  -0.868   0.061  -0.207  -0.121  -0.889   0.364   0.621   0.493   0.186  -0.079  -0.281
        -0.619  -1.166
4:     -14.234  -5.737  -4.246  -4.206   2.187  -3.432  -0.626  -0.092  -2.288  -1.455   0.348  -1.702  39.203  -0.394  -0.004   0.253  -0.837  -1.114  -0.216
   0.096   0.975   1.462   0.768   0.454  -1.492   0.905  -0.398   0.295  -0.023   0.382  -0.576   0.502   0.578  -0.303   0.273  -0.220  -0.365
        -0.185  -0.989
5:     -15.205  -4.162  -4.429  -6.349  -2.095  -3.353   0.671   0.292   1.345   0.751   4.535   1.280  39.704  -0.278  -0.176  -0.440  -0.574  -1.969   0.264
   1.019   0.255   1.266   0.287  -0.439  -1.723   0.685   0.097   0.189   0.074   0.769   0.368   0.366   0.221  -0.691  -0.044  -0.168  -0.256
          0.296  -0.422
6:     -14.500  -3.604  -1.831  -3.600  -3.936  -0.148   1.491  -3.950   5.037  -1.709  -2.209  -5.999  40.126   0.206   0.474   0.305   0.713  -0.790  -0.067
   1.272  -0.475   1.885  -0.326  -0.872  -0.388   0.632   0.133   0.044   0.010   0.452   0.871  -0.065   0.007  -0.150  -0.414  -0.163   0.011
          0.864   0.009
7:     -14.781  -4.957  -5.301  -4.189  -3.598  -3.202   1.800  -1.978   2.933  -0.074  -0.516  -3.462  40.780   0.281  -0.026   0.210   1.531   1.462  -0.006
   0.526  -0.610   0.671   0.291  -0.138   0.622   0.780   0.082   0.066   0.201   0.011   0.627  -0.062  -0.323   0.291  -0.539  -0.118   0.062
          0.621   0.065
8:     -13.416  -2.968  -2.287  -1.723  -1.015  -3.840   5.171  -1.331   6.343  -2.673  -1.487  -1.271  41.829  -0.010   0.142  -0.020   0.370   1.524  -0.408
   0.379   0.657  -0.311  -0.051   0.360   1.658   0.902  -0.029  -0.044   0.062  -0.541  -0.259   0.045  -0.466   0.463  -0.725  -0.189   0.081
          0.027   0.022
```

Figure 4. Screenshot of an MFCC file.

A prototype model is shown in **Figure 5**.

These models were further refined by applying nine iterations of the standard Baum-Welch embedded training procedure. These models are then converted to tri-phone models and two iterations of Baum-Welch training procedure are applied, then the states are tied using decision tree based approach and iterations of Baum-Welch training procedure are applied. **Figure 6** shows the training procedure.

4. Evaluation Methodology

The performance of the ASR is tested while transcribing unknown utterances. A database which is not used for training the system is called unseen data. The test data here is an exclusive set consisting of 344 unseen utterances spoken by 8 speakers (4 males and 4 females) each speaking 43 sentences.

```
~o
<STREAMINFO> 1 39
<VECSIZE>
39<NULLD><MFCC_D_A_0><DIAGC>
~h "proto
<BEGINHMM>
<NUMSTATES> 5
<STATE> 2
<MEAN> 39
-7.055892e+00 -2.760827e+00 -1.855420e+00 …….
<VARIANCE> 39
3.614088e+01 4.895053e+01 6.375173e+01 …….
<GCONST> 1.185559e+02
<STATE> 3
<MEAN> 39
-7.055892e+00 -2.760827e+00 -1.855420e+00 ….
<VARIANCE> 39
3.614088e+01 4.895053e+01 6.375173e+01 …….
<GCONST> 1.185559e+02
<STATE> 4
<MEAN> 39
-7.055892e+00 -2.760827e+00 -1.855420e+00 ……
<VARIANCE> 39
3.614088e+01 4.895053e+01 6.375173e+01 …….
<GCONST> 1.185559e+02
<TRANSP> 5
0.000000e+00 1.000000e+00 0.000000e+00
0.000000e+00 0.000000e+00
0.000000e+00 6.000000e-01 4.000000e-01
0.000000e+00 0.000000e+00
0.000000e+00 0.000000e+00 6.000000e-01
4.000000e-01 0.000000e+00
0.000000e+00 0.000000e+000.000000e+00
7.000000e-01 3.000000e-01
0.000000e+00 0.000000e+00 0.000000e+00
0.000000e+00 0.000000e+00
<ENDHMM>
```

Figure 5. A prototype model.

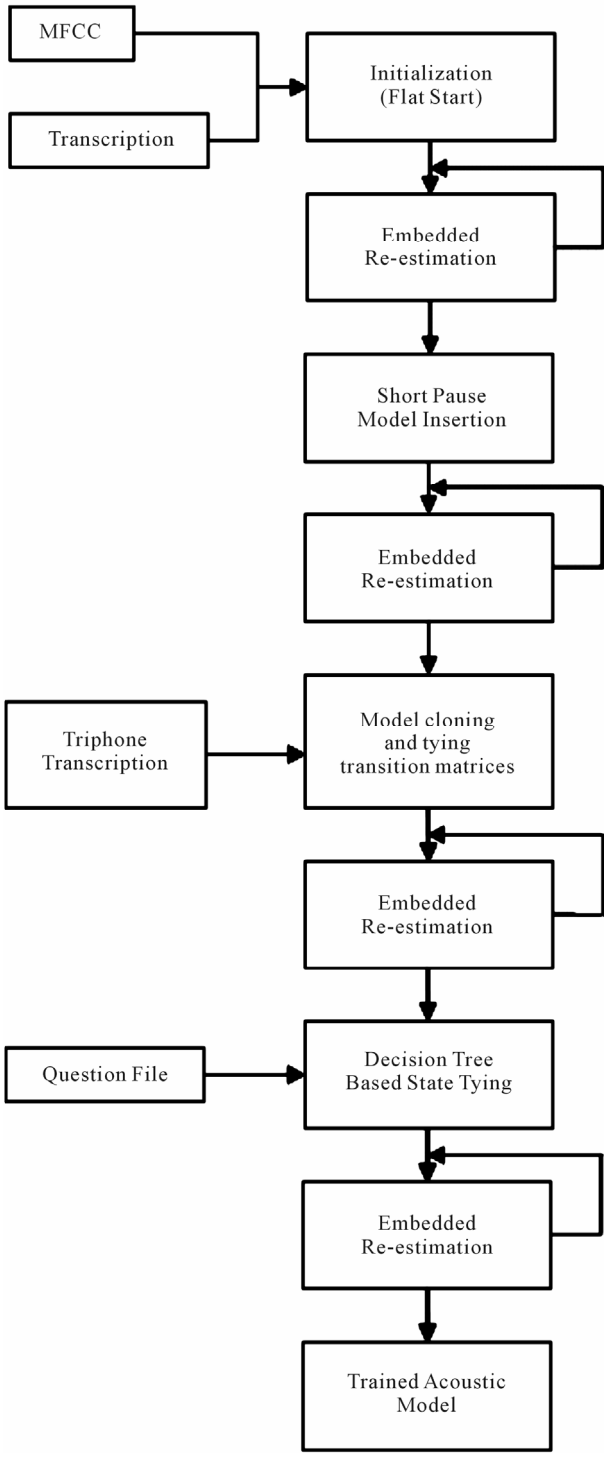

Figure 6. Acoustic model training methodology.

4.1. Role of Language Model

In speech recognition the Language Model is used for the task of finding word boundaries, that is, segmentation. The language model or grammar which is an *a priori*

knowledge of the syntax, semantics and pragmatics of the language in question, helps decode the sequence of phonemes into different words in a sentence. An example is given in **Figure 7**.

Here the constraints applied by the Language model helps the Recognizer in decoding the phoneme sequence into words. We have generated our own language model.

4.2. The Recognizer

The decoder used for recognition is Julius. Since Julius itself is a language-independent decoding program [23], we can make a recognizer of any language if given an appropriate language model and acoustic model for the target language. The recognition accuracy largely depends on the models. Julius is a real-time, high-speed, accurate recognition engine based on 2-step strategy. It works in two steps. The first step is a high-speed approximate search, which uses a 2-gram frame synchronous beam searching algorithm. In the first step, a tree-structured lexicon assigned with the language model probabilities was applied. Pre-computed unigram factoring values are assigned to the intermediate nodes and bi-gram probabilities on the word-end nodes.

The second step is a high precision trigram N-best stack decoding. The tree trellis search in the second pass recovers the degradation caused by the rough approximation in the first step. Julius adopts acoustic models in HTK ASCII format, pronunciation dictionary in almost HTK format, and word 3-gram language models in ARPA standard format (forward 2-gram and reverse 3-gram trained from same corpus). The following is a sample output for one of our test utterances.

4.3. The Evaluation Parameters

Finally, the recognition accuracy of the Speaker Independent ASR system and the percentage of correct words and percentage of correct sentences were calculated using the following formulae.

$$\%\text{correct} = H/N \tag{12}$$

where, H = Number of labels (sentences here) correctly recognized

N = Total number of labels

$$\%\text{Recognition Accuracy} = \left(N - D - S - I\right)/N \tag{13}$$

D = Number of unrecognized/missed words. (Deletion errors)

S = Number of times a word was misrecognized as another word (Substitution errors)

I = Number of extra words inserted between correctly recognized words (Insertion errors)

N = Total number of words or sentences

[*ey k sp s ax m aa n t ax r sp ch ax t uh r b hh uh jh sp b ax n aa ow sp*]
एक समान्तर चतुर्भुज बनाओ

Figure 7. The Recognition of series of phonemes into series of words.

5. Results and Discussion

The system was trained with 1806 Hindi utterances (sentences) spoken by 18 males and 12 females. The performance of the system was evaluated both for seen and unseen speech data. All the 43 distinct sentences for which the system was trained were uttered by 8 persons (4 males and 4 females). A total of 316 test (unseen) utterances and 1371 seen utterances were used in testing. The % of correct sentences and Recognition Accuracy were calculated using formulae given above and results are shown in **Table 2**.

The Recognition Accuracy for males is better as expected as the ASR is speaker independent and the male speech data is more than that of females. The amount of training data must be increased to achieve better speaker independent model.

6. Conclusion and Future Work

We have proposed an approach to implement a continuous speech recognition system in Hindi customized for computer aided teaching of geometry. We have used the MFCC as speech feature parameters and HMM to model the acoustic features. HTK-3.4 was used both for feature extraction and model generation. The Julius recognizer which is language independent was used for decoding.

The present work was limited to 29 phonemes of Hindi. It is mostly demonstrative in nature. The future endeavor will be to make the system full-fledged by increasing vocabulary to include all required words and all the Hindi phonemes. A phonetically balanced and rich database for the said application will be created and used.

More training data will be collected and used to improve the speaker Independent system. Methods to improve the Recognition rate of the speaker Independent

Table 2. Recognition accuracies and % of correct words for speakers in training and test sets.

Speakers		% Correct Sentences	% Recognition Accuracy (Words)
Training (Seen)	Male (18)	76.84	92.72
	Female (12)	60.28	84.9
	All (30)	68.56	88.81
Test (Unseen)	All (8 = 4M + 4F)	42.72	79.11

system will be studied and experimented. Feature sets other than MFCC will be tested for reducing speaker and other variability

REFERENCES

[1] K. Kumar and R. K. Agarwal, "Hindi Speech Recognition System Using HTK," *International Journal of Computing and Business Research*, Vol. 2, No. 2, 2011, ISSN (Online): 2229-6166.

[2] G. Sivaraman and K. Samudravijaya, "Hindi Speech Recognition and Online Speaker Adaptation," *Proceedings of ICTSM* 2011, Vol. 145, 2011, pp. 233-238.

[3] D. ShakinaDeiv, Gaurav and M. Bhattacharya, "Automatic Gender Identification for Hindi Speech Recognition," *International Journal of Computer Applications*, Vol. 31, No. 5, 2011, pp. 1-8.

[4] R. K. Aggarwal and M. Dave, "Implementing a Speech Recognition System Interface for Indian Language," *Proceedings of the IJCNLP*-2008 *Workshop on NLP for Less Privileged Languages*, Hyderabad, January 2008, pp. 105-112.

[5] R. Mathur, Babita and A. Kansal, "Domain Specific Speaker Independent Continuous Speech Recognition Using Julius," ASCNT 2010.

[6] S. Arora, B. Saxena, K. Arora and S. S. Agarwal, "Hindi ASR for Travel Domain," Oriental COCOSDA 2010 *Proceedings Centre for Development of Advanced Computing*, Noida, 24-25 November 2010.

[7] R. K. Aggarwal and M. Dave, "Fitness Evaluation of Gaussian Mixtures in Hindi Speech Recognition System," 2010 *First International Conference on Integrated Intelligent Computing*, Bangalore, 5-7 August 2010, pp. 177-183. doi:10.1109/ICIIC.2010.13

[8] K. Samudravijaya, "Hindi Speech Recognition," *Journal Acoustic Society of India*, Vol. 29, No. 1, 2009, pp. 385-393.

[9] K. Malhotra and A. Khosla, "Automatic Identification of Gender & Accent in Spoken Hindi Utterances with Regional Indian Accents," *IEEE Spoken Language Technology Workshop*, Goa, 15-19 December 2008, pp. 309-312.

[10] R. Gupta, "Speech Recognition for Hindi," M. Tech. Project Report, Department of Computer Science and Engineering, Indian Institute of Technology, Bombay, Mumbai, 2006.

[11] B. A. Q. Al-Qatab and R. N. Ainon, "Arabic Speech Recognition Using Hidden Markov Model Toolkit (HTK)," *International Symposium in Information Technology*, Kuala Lumpur, 15-17 June 2011, pp. 557-562.

[12] C. Kurian and K. Balakrishnan, "Speech Recognition of Malayalam Numbers," *World Congress on Nature & Biologically Inspired Computing*, Coimbatore, 9-11 December 2009, pp. 1475-1479.

[13] R. Syama and S. M. Idikkula, "HMM Based Speech Recognition System for Malayalam," *The International Conference on Artificial Intelligence*, 2008Monte Carlo Resort, Las Vegas, 14-17 July 2008.

[14] P. G. Deivapalan and H. A. Murthy, "A Syllable-Based Isolated Word Recognizer for Tamil Handling OOV Words," *The National Conference on Communications*, Indian Institute of Technology Bombay, 1-3 February 2008, pp. 267-271.

[15] C. Neti, N. Rajput and A. Verma, "A Large Vocabulary Continuous Speech Recognition System for Hind," *IBM Research and Development Journal*, September 2004.

[16] G. Anumanchipalli, R. Chitturi, S. Joshi, R. Kumar, S. P. Singh, R. N. V. Sitaram and S. P. Kishore, "Development of Indian Language Speech Databases for Large Vocabulary Speech Recognition Systems," *Proceedings of International Conference on Speech and Computer* (*SPECOM*), Patras, October 2005.

[17] A. Stolcke, "SRILM—An Extensible Language Modeling Toolkit," *Proceedings of the 7th International Conference on Spoken Language Processing*, 2002, pp. 901-904. http://www.speech.sri.com/

[18] L. R. Rabiner, "A Tutorial on Hidden Markov Models and Selected Applications in Speech Recognition," *Proceedings of the IEEE*, Vol. 77, No. 2, 1989, pp. 257-286.

[19] C. J. Legetter, "Improved Acoustic Modeling for HMMs Using Linear Transformations," Ph.D. Thesis, University of Cambridge, Cambridge, 1995.

[20] M. Benzeghiba, R. De Mori, O. Deroo, S. Dupont, T. Erbes, D. Jouvet, L. Fissore, P. Laface, A. Mertins, C. Ris, R. Rose, V. Tyagi and C. Wellekens, "Automatic Speech Recognition and Speech Variability: A Review," *Speech Communication*, Vol. 49, No. 10-11, 2007, pp. 763-786. doi:10.1016/j.specom.2007.02.006

[21] T. Herbig, F. Gerl, W. Minker and R. Haeb-Umbach, "Adaptive Systems for Unsupervised Speaker Tracking and Speech Recognition," *Evolving Systems*, Vol. 2, No. 3, 2011, pp. 199-214. doi:10.1007/s12530-011-9034-1

[22] Steve Young, *et al.*, "The HTK Book," http://htk.eng.cam.ac.uk/docs/docs.shtml

[23] A. Lee, T. Kawahara and K. Shikano, "Julius—An Open Source Real-Time Large Vocabulary Recognition Engine," *Proceedings of 7th European Conference on Speech Communication and Technology*, 2001.

