Speaker Independent Speech Recognition of English Digits using Hybridized VQ-HMM Model in Noisy Environment

P. Prithvi
Department of ECE
National Institute of Technology,
Warangal, India

Anil Kumar
Department of ECE
National Institute of Technology,
Warangal, India

Dr. T. Kishore Kumar
Department of ECE
National Institute of Technology,
Warangal, India

Abstract— **This paper provides an analysis of recognition rate of English digits ("one" to "ten") in noisy environment. Automatic Speech Recognition (ASR) is not a new topic, but when deal with noisy environment and speaker independent recognition system, then it's requires lot of improvement. ASR accuracy is maximized by maximizing the word recognition rate. In this paper, pattern recognition approach is used i.e. based on template matching. Hidden Markov Models is one of the best pattern recognition approaches. In previous work, recognition rate is 67% for speaker independent in noisy environment. This paper is based on hybridized model of Vector quantization and Hidden Markov Model. Vector quantization is used to quantize the data vector within certain limit using K-Means algorithm. The Mel-frequency cepstral coefficients (MFCC) are used as a feature extraction approach to extract the feature of input analog speech signal. Experimental results show the improvement in recognition rate is up to 81.8 % with proposed hybridized VQ-HMM model.**

Keywords-Automatic Speech Recognition (ASR) System, Mel Frequency Cepstral Coefficients (MFCC), Vector Quantization (VQ), Hidden Markov Model (HMM)

I. INTRODUCTION

Recognizers play very important role for performance of ASR system. A long time work is going on speech recognition but still the complete solution is far away from destination. Recent work gave good result for recognition of voice in clean environment but when deal with noisy environment, there is more improvement require in recognition rate [1]. Speech recognition is the process to recognize the word spoken by speaker. Design of system is depend on application either it is related to specific speakers or independent speakers. Speech recognition is used in those instruments which can operate on voice commands. There are various methods available for speech recognition [5]. Pattern recognition approach is one method for ASR i.e. based on template matching. Pattern recognition is based on maximum likelihood ratio [7]. In this method, two steps are involved: training of word pattern and pattern comparator [5].

Objective of this paper is to design automatic speech recognition system that recognized the English digit spoken by any person in noisy environment. English digits are recognized by the system which is laid in the range of one to ten. Before pattern recognition, signal modeling is required to extract the features of input signal. For recognition of same digit, record many no utterances in clean and noisy environment both. Each utterance is recorded by different speaker because system is trained for speaker independent [1]. As per fig.1 isolated word recorded by the microphone is passed to feature extraction block. Feature extraction is the part of signal modeling. Mel-frequency cepstral coefficient is used to extract the features of input speech signal. Extracted feature is passed to next block for codebook generation. It quantizes the frames of each utterance within limited clusters and generates the codebooks.

Input testing speech signal is an unknown signal recorded by any speaker. It is not necessary that training and testing speaker should be same. The task of pattern comparator is to find best matching of input testing speech with prerecorded data. Hidden Markov Model is used to compare the testing speech signal with parameters trained by previous recorded data. Testing speech signal is compared with each model and gives respected likelihood value. Maximum likelihood value among all the comparison is treated as the best suited model for the testing word [7]. Word correspond to that suited model is recognized as unknown testing word.

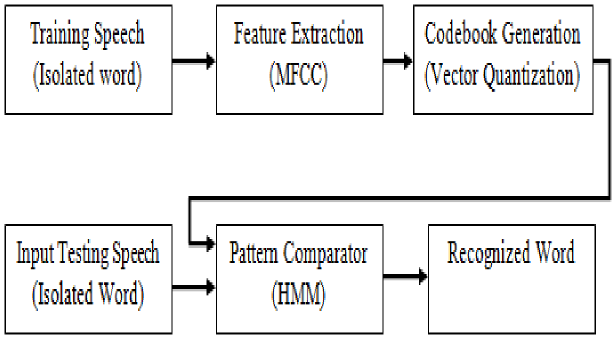

Fig. 1. Block Diagram of automatic speech recognition

II. INPUT SPEECH

Speech signal is captured by microphone in clean or noisy environment that converts variable sound pressure into equivalent voltage. Speech signal is recorded by different variable to make the system speaker independent. Some noise is added into signal due to microphone, A/D convertor and background environment. For higher performance these type of noise should be eliminated before feature extraction.

Speech signal is also attenuated at high frequency. This attenuation can be compressed be using pre-emphasis high pass filter [4].

III. FEATURE EXTRACTION

Feature extraction is applied on signal to extract the information from signal. Mel frequency cepstral coefficient (MFCC) is very helpful to extract the features of speech signal. Speech signal is a non-stationary signal. First convert the non-stationary signal into stationary. For a small duration of time (<20 ms), it can be assume that speech signal is stationary. Speech signal is divided into constant length of frames with overlapping. Each frame consists of samples less than 20 ms duration. Overlapping is provided for smooth transition of signal from frame to frame. Each frame is windowed (1) with hamming window to decreases the discontinuity and energy at the edges of each frame [4]. The filter coefficients w(n) of Hamming window of length n are computed as,

$$w(n) = \begin{cases} 0.54 - 0.46 * \cos\left(\frac{2 * \pi * n}{P-1}\right) & for\ 0 \le n \ge P-1 \end{cases} \quad (1)$$

Where P is total no of samples in each frame and n is current sample. After windowing, Fast Fourier Transform is computed to convert each frame from time domain to frequency domain. N-point FFT is used to speed the processing. Zero padding is used to adjust the frame length of windowed signal before applying N-point FFT.

Human ear is very sensitive to frequency above 1 kHz and also it is not follow a linear scale above 1 kHz. Pitch of human speech is measured on Mel scale. So Mel scale should be linear below 1 kHz and logarithmic above 1 kHz [6]. Using (2) Mel scale can be computed for a given frequency.

$$frequency\,(mel\,scale) = \left[2595 * \log\left(1 + \frac{f(Hz)}{700}\right)\right] \quad (2)$$

Mel scale is used to design the overlapped triangular band pass filter i.e. is also called Mel-scale filter bank [4]. The centre frequency of filter bank is according to Mel scale, means below 1 kHz filters are linearly spaced and above 1 kHz filters are logarithmically spaced. Filters below 1 kHz are called linear filters and above 1 kHz is called logarithmic filters. Linear filters have same bandwidth but logarithmic filter's bandwidth is increasing on increasing the frequency.

Magnitude spectrum obtains from Fourier transform of each frame is mapped on Mel-scale filter bank. Filter bank gives Mel spectrum for each frame. Logarithmic is applied on Mel spectrum to calculate log-spectral-energy for each frame [8].

The final step of MFCC is to obtain the mel cepstrum which is a time domain representation of spectral properties of the signal [3]. Mel-frequency cepstral coefficients (MFCCs) are time domain coefficients. Discrete Cosine Transform is applied on log-spectral-energy to compute the mel cepstrum for each frame. For each utterance, a set of MFCC is computed i.e. also called feature vectors or data vector. MFCC of each utterance gives the matrices of [N X M], where N represent total no of MFCC coefficients & M represent total no of frames. Each frame is considered as N-dimensional data vector.

Fig.2 shows the steps involved in computation of MFCC coefficients.

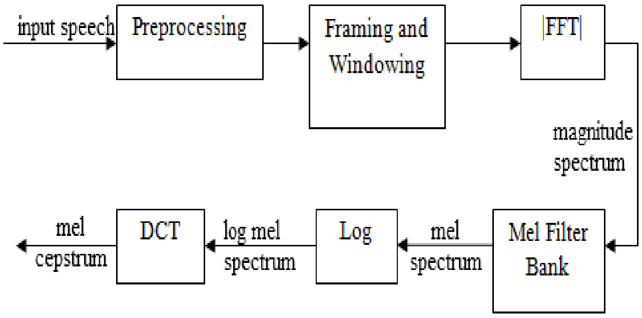

Fig. 2. Block Diagram of Mel-Frequency Cepstral Coefficient

IV. VECTOR QUANTIZATION

Due to large number of feature vectors of all utterances the amount of computation significantly increases. Before training of each word, feature vector should be reduced. So vector quantization is applied after feature extraction of each word. The goal of vector quantization is to quantize the frames of each utterance within limited clusters and the resulted vector is called codebook vector [4].

K-means algorithm is used for vector quantization. Cluster size (K) should be less then total no of frames [4]. All frames should be lie within the K clusters. Clusters can be represent by their index number (1,2,3,…K). According to K-means algorithm first choose K random vectors from the data vectors, that random vector is defined as codebook vector and set of all codebook vectors is called codebook. First each codebook vector is assign in respective index cluster number. Calculate the distortion measure between code vector and data vector. For the optimal codebook, distortion measure should be minimum value [2] [3]. The squared Euclidian distance (3) is very helpful to calculate the distortion measure. Calculate the squared Euclidian distance for each data vector from each code vector.

Let x represent a single data vector and y_i is code vectors, where y_1 represent first code vector and so on. Compute the squared Euclidian distance for x from each code vector.

$$d(x, y_i) = \sqrt{\sum_{j=1}^{j=N} \left(x^j - y_i^j\right)^2} \qquad where\ i = 1,2,3...K \quad (3)$$

Assign each data vector in that particular code vector cluster from where that data vector has minimum squared Euclidian distance.

After assigning data vectors into clusters, apply the centroid theorem for each clusters using (4),

$$Centroid\,(C_i) = \frac{\sum_{x_m \in C_i} x_m}{\sum_{x_m \in C_i} 1} \qquad for\ i = 1,2,....K \quad (4)$$

Centroid of each cluster is assign as new code vector for that cluster. This process is repeated, until the difference between the new code vectors and old code vector is sufficiently small.

Individual Code book is generated for each utterance and that codebook is called observation symbol for next step.

V. HIDDEN MARKOV MODEL (HMM)

Next step is pattern recognition. HMM is used as pattern recognition approach to recognize the words [5]. In pattern recognition, two steps are involved namely training of word pattern and pattern comparison.

Hidden Markov Model can be represented as a finite state machine which has unknown states and observation sequences. Let *HMM* model have S states and K observation symbol in each state. Assume that the observation sequence of each utterance obtain from vector quantization is *O*. HMM use three parameters [10]:

1) Transition Probability (A): It is defined as probability of change of state from time t to time t+1. For each transition, present state can be change to any one of S state. So it can be represented as [S X S] matrix.

2) Observation Probability (B): It is defined as probability of generating any one of observation symbol in present state. So in particular state, it generates one symbol out of K symbols. It can be represented as [K X S] matrix.

3) Initial State Probability (π) *:* It is defined as probability of having in certain state as initial state. Initially it can be in one state out of S state. So it can be represented as [S X 1] matrix.

Hidden Markov Model can be represented as a set of parameters $\lambda = (A, B, \pi)$. These parameters are trained according to observation sequence of each word.

The three basics problem associated with HMM must be solved for the real world application [5] [9]:

1) Evaluation: Calculate the probability P(O/λ) for producing the given observation sequence from set of parameters λ.

2) Decoding: Find best state sequence from set of parameters for given observation sequence.

3) Training: Adjust the model parameter λ so that it gives maximum probability to generate that observation sequence.

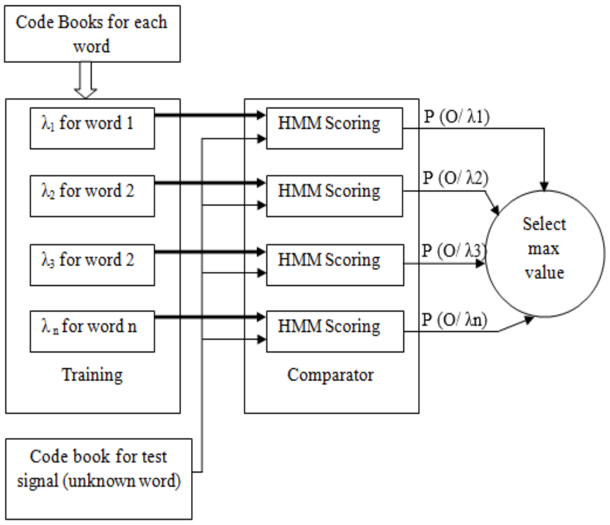

Fig. 3. Block Diagram of pattern recognition using HMM

Forward-Backward algorithm is used to solve these three problems [10]. After training of word pattern, model parameter (λ) is trained according to training data. Now testing data is recorded as an unknown signal for recognition. MFCC and Vector Quantization generate the observation sequence for testing data. Observation sequences (O) is passed to pattern comparator and compare with model parameter and give likelihood value with respect to each digit's model parameter. Digit correspond to maximum likelihood value is treated as recognized digit.

Fig.3 represents the process of pattern recognition using HMM.

VI. EXPERIMENTAL RESULTS

The complete programming was done in MATLAB. Input digits were recorded by microphone in both noisy and clean environment. For recording the digits from different speaker, MATLAB program was used instead of system recorder to adjust the sampling frequency and duration of speech. This system has 10 digits ("one" to "ten") and each digit was recorded 3 times by 10 different speakers. For each digit, 30 utterances were recorded. Total 300 utterances were recorded for training and save as .wav format in memory. Each utterance was recorded for 2 second at 8000 sampling frequency. Table I showed the data used in this experiment and their respective values.

TABLE I. SUMMARY OF DATA

Data	Values
Frame Size	80 sample
Total no of Frame	267 frames
Hamming Window Length	80 sample
FFT	128 point FFT
Mel-Filter Bank	40(13 liner filters and 27 log filters)
Cluster Size	50
States (S)	5
Observation symbol (K)	50

HMM parameter $\lambda = (A, B, \pi)$ is trained for each digit. For testing the experiment, each digit was recorded 5 times by 10 different speakers. Total 50 utterances were used for each digit in testing. The codebook was generated for each testing utterance. Maximum likelihood was calibrated for each testing utterance. Digit correspond to that maximum likelihood was treated as recognized digit. This process was repeated for all testing utterances. Calculate the recognition rate for each digit. Recognition rate *(R R)* (5) is defined as total no of recognized utterances $(N_{recognized})$ for each digit to the total no of utterances (N_{total}) for each digit.

$$RR = \frac{N_{recognized}}{N_{total}} \times 100 \qquad (5)$$

Table II show the recognition rate for each digit in clean and noisy both environment. Average recognition rate is 92.6 % in clean environment and 81.8 % in noisy environment for speaker independent mode. Table III show the comparison of recognition rate in noisy environment with previous result [1].

TABLE II. RECOGNIZED RATE (%) FOR EACH DIGIT IN SPEAKER INDEPENDENT MODE

Words	**Recognition Rate (%)**	
	Clean Environment	***Noisy Environment***
"One"	96	86
"Two"	96	84
"Three"	92	80
"Four"	94	82
"Five"	92	80
"Six"	90	84
"Seven"	88	76
"Eight"	90	80
"Nine"	92	78
"Ten"	96	88
Average Recognition Rate	**92.6**	**81.8**

TABLE III. COMPARISON OF RECOGNITION RATE (%) IN SPEAKER INDEPENDENT MODE WITH PREVIOUS RESULT

Reference	**Features Extraction Techniques**	**Recognition Techniques**	**Recognition Rates (%)**
[1]	MFCC	HMM	79.5 (Clean) 67 (Noisy)
From Table II	MFCC	HMM+ VQ	92 (Clean) 81.8 (Noisy)

VII. CONCLUSION

The goal of this paper was to recognize English digits ("one" to "ten") in noisy environment. The microphone recorded each utterance in clean and noisy environment both. MFCC algorithm was used for feature extraction of input data. Vector Quantization was used to correlate utterances of different speaker for same word and generated the codebook. HMM was used for training of parameter and pattern comparator. After recognition of word it was found that there is trade-off between recognition and environment. Recognition rate is increased up to 81.8% in noisy environment for speaker independent mode. Recognition rate was high for clean environment as compare to noisy environment. But as compare to previous result, the recognition rate is high for noisy environment.

REFERENCES

[1] Ahmed A.M. Abushariah, Teddy S. Gunawan, Othman O. Khalifa and Mohammad A.M. Abushariah, "English digits speech recognition system based on Hidden Markov Models," International Conference on Computer and Communication Engineering(ICCCE), 2010.

[2] Hiren Parmar and Bhagwan Sharma, "Control system with speech recognition using MFCC and Euclidian Distance Algorithm," International Journal of Engineering Research & Technology(IJERT), vol.2 issue 1, January-2013

[3] Vibha Tiwari, "MFCC and its application in speaker recognition," International Journal on Emerging Technology, 2010.

[4] Dipmoy Gupta, Radha MounimaC., Navya Manjunath and Manoj PB, "Isolated word speech recognition using Vector Quantization (VQ)," International Journal of Advanced Research in Computer Scoence and software Engineering, vol.2 issue 5, May 2012.

[5] Bhupinder Singh, Neha Kapur and Puneet Kaur, "Speech Recognition with Hidden Markov Model: A Review," international Journal of Advanced Research in Computer Science and Software Engineering, vol.2, issue 3, march 2012.

[6] Ibrahim Patel and Dr. Y. Srinivasa Rao, "Speech recognition using Hidden Markov Model with MFCC-subband technique," International Conference on Recent Trends in Information, Telecommunication and Computing, March 2010, pp.168-172.

[7] Prof. Ashok Shigli, Ibrahim Patel and Dr. K. Srinivas Rao, "A spectral feature process for speech recognition using HMM with MFCC approach," National Conference on Computing and Communication System (NCCCS), Nov 2012.

[8] Shweta Tripathy, Neha Baranwal and G.C.Nandi, "A MFCC based Hindi speech Recognition Technique using HTK toolkit," IEEE Second international Conference on Image Information Processing(ICIIP-2013), Dec-2013, pp.539-544.

[9] V. Amudha, B. Venkataramani, R.Vinoth Kumar and S. Ravishankar, "SOC implementation of HMM Based Speaker independent isolated digit recognition system," 20th International Conference on VLSI Design (VLSID'07), 2007, pp.848-853.

[10] Behroz Vaseghi, Shahpour Alirezaee, Majid Ahmadi and Rasoul Amirfattahi, "Off-line Farsi/Arabic handwritten Word Recognition Using Vector Quantization and Hidden markov Model," IEEE International on Multitopic Conference, 2008.