**International Journal of Pure and Applied Mathematics**
**Volume 119 No. 12 2018, 15011-15024**
ISSN: 1314-3395 (on-line version)
**url:** http://www.ijpam.eu
Special Issue

ijpam.eu

# Multi-Level Processing for Continuous Speech Recognition in Natural Environment

[1]Praveen Edward James, [2]Mun Hou Kit, [3]Chockalingam Aravind Vaithilingam and [4]Alan Tan Wee Chiat

[1]School of Engineering,

Taylor's University, Taylor's University Lakeside Campus, No. 1,

Jalan Taylor's, Subang Jaya, Selangor, Malaysia.

praveenedwardjames@sd.taylors.edu.my

[2]School of Engineering,

Taylor's University, Taylor's University Lakeside Campus, No. 1,

Jalan Taylor's, Subang Jaya, Selangor, Malaysia.
HouKit.Mun@taylors.edu.my

[3]School of Engineering,

Taylor's University, Taylor's University Lakeside Campus, No. 1,

Jalan Taylor's, 47500 Subang Jaya, Selangor, Malaysia.
ChockalingamAravind.Vaithilingam@taylors.edu.my

[4]School of Engineering,

Multimedia University, Melaka.

wctan@mmu.edu.my

## Abstract

In a natural environment, the performance of a Hidden Markov Model (HMM) based speech recognition system degrades with noise. To overcome this limitation, additional processing techniques are required. This paper involves the design of sentence recognition system with multi-level processing techniques like Dynamic Time Warping (DTW) and Multi-Instance Training (MIT). This entire process involves two phases: training and recognition. The training phase of speech recognition involves preprocessing the signal to eliminate noise by band pass filtering, pre-emphasis, Voice Activity Detection (VAD), DTW and MIT Training. In the recognition phase the test sentences are processed into individual words

and compared with training data and the recognized words are presented as text. Two stages of experiments are performed. In the first stage, a measure of accuracy called the Word Error Rate (WER) is calculated for the stand-alone system and estimated as 24.1. In the second stage, DTW is performed during preprocessing and MIT during training in a sequence of steps and the WER for each step was obtained as 23.6 and 23.4 respectively. The result shows that there is a 2.9% decrease in WER with preprocessing the basic system with DTW and MIT based training. The system has strong adherence to mathematical concepts and hence it is reliable, stable and suited for resource constrained environments

**Keywords:** Speech, Recognition, feature extraction, Cepstral coefficients, Pre-emphasis, Estimation.

# 1. Introduction

Speech recognition is an extensively explored area in speech processing. It has evolved from being a biometric identification tool, to control small devices like smart phones and larger machines like automobiles. In natural environments like dining in a restaurant, work place and driving an automobile, the performance of the system can be degraded by noise interference. Some of the existing applications available commercially are Voice Search (Google), Siri (Apple), Cortana (Microsoft) and so on [1]. Most of them are a combination of stand-alone and server based applications. Some of the common problems with internet connectivity will occur in these devices.

In this paper some of the sentences commonly used in smart devices are used to train a HMM based continuous word speech recognition system. A HMM uniquely represents and accurately models a speech signal and in combination with pre-processing techniques an upgraded system is designed. By comparing the performance of the systems with and without preprocessing techniques in in a natural environment the results can be analyzed and performance can be deduced.

A justification for deploying the HMM model is achieved by a review of some of the recent work. A HMM is a finite state machine which transits between a finite number of states and generates an observation during each transition [2]. There are many types of HMM including an ergodic HMM where there can be transitions from any states, a simple left-to-right HMM where there is a fixed starting and ending state with possible transitions from left to right or back to the same state and an autoregressive model. A left- to- right HMM with 5 states are shown below (Fig 1.1) is very apt for creating speech recognition models.

In the Speech recognition model speech is acquired as a sentence and later split into words by repeatedly matching their features with stored templates. The acoustic features are captured using Mel Frequency Cepstral Coefficients (MFCC) based feature extraction. The performance of the phrase recognition model is accessed by the Word Error Rate (WER). In late 1960s, Rabiner [2] introduced Hidden Markov Models (HMM) for applications in speech recognition.

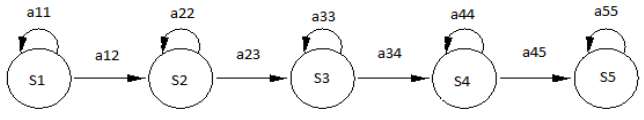


Figure 1.1A five states HMM [3]

Hinton *et al.,* observed that HMM based recognition in combination with Gaussian Mixture Model (GMM) has maximized performance in linear manifold [3]. An efficient connected digit recognizer was modeled by Srichai [4], by using Viterbi algorithm for both training and recognition.

Hoang *et al*., [5] proposed an algorithm for increasing the speed of decoding by implementing a hardware model. Tan *et al*., [6] compared different classifiers in a subspace. Gales and Young [7] used HMM for designing a Large Vocabulary Continuous Speech Recognition system. Kepuska and Elharati [8] proposed a hybrid feature extraction process for HMM in noisy conditions. Mine*et al.,* [9] designed a low-bandwidth recognition system using Celoxica RC250 FPGA Board.

He *et al*., [10] proposed a high speed large vocabulary system with low power consumption implemented using VLSI technology. All the above models are stable and reliable due to their adherence to mathematical concepts. This paper involves the design of a speech recognition system using HMM that can operate in natural environments.

# 2. Research Method

The design of the speech recognition involves speech acquisition, band pass filtering, pre-emphasis, end-point detection (EPD), Dynamic Time Warping (DTW), feature extraction (FE), creation of HMM models, parameter estimation (PE) and sentence recognition. The entire process can be divided into training and testing phases.

### 2.1 Training Phase:

Figure 2.1 shows the stages involved in the training phase.

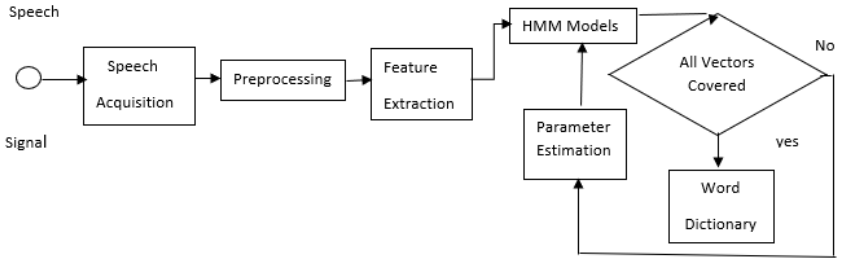


Figure 2.1 Stages involved in the training phase.

### 2.1.1. Speech Acquisition

Speech acquisition is performed at two-levels namely the sentence-level and word-level. At the word-level a training set of the words that constitute ten sentences selected for testing are spoken and converted into .wav files. The signals are recorded using windows direct sound. Each signal has a sampling rate of 16 kHz and stored with a precision of 16 bits/sample. During the test phase the entire waveform is captured and is shown in Figure2.2.

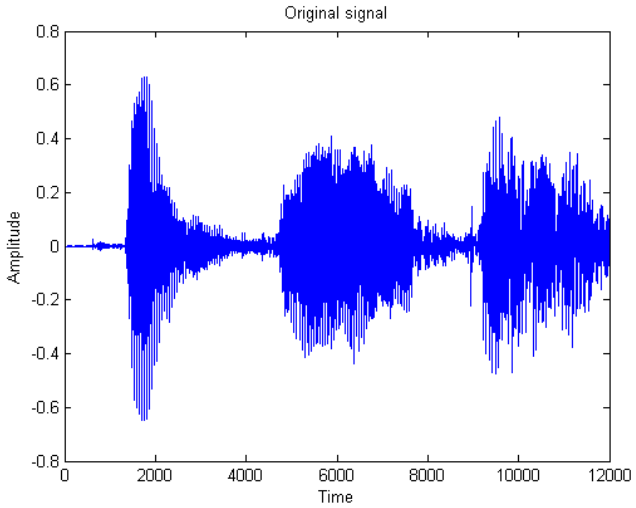


Fig. 2.2. Original Speech Signal (Good Morning Praveen)

### 2.1.2 Noise Cancellation:

Noise cancellation targets ambient noises that are mixed with the speech signal in a home or office environment [5]. Since it involves both low frequency and high frequency signals, a band pass filter is used. The occurrence of low frequency noise is also considered. The digitized speech signal that has a sampling frequency of 16 kHz is passed through a band pass filter with a minimum frequency of 2000 Hz and maximum frequency of 8000 Hz. The resulting waveform is shown in Figure 2.3.

### 2.1.3 Pre-emphasis

This involves a high-pass filter to increase the amount of energy in the high frequencies and enabling easy access to information in the higher formants of the speech signal

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

where $\ddot{a}$ is 0.95 and $H(z)$ is the filter function expressed in a Z-transform representation. As the order of the filter is increased the precision is increased at the expense of omitting some signals with unique characteristics. The value of $a$ at which the high frequency energies are maximum is chosen. The output signal of a pre-emphasis filter is shown in Figure 2.4.

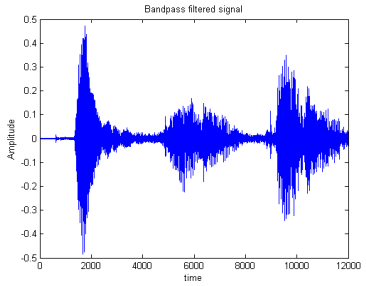


Fig.2.3. Band-Pass filtered Signal

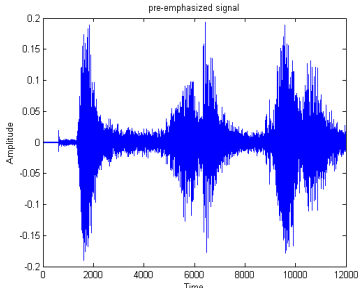


Fig. 2.4. Pre-emphasized Signal

### 2.1.4 Voice Activity Detection (VAD)

VAD is also referred as end-point detection. Initially the signal is divided into equal frames. Using the sum of squares energy formula given in Eq. (2), the energy of the speech signal is calculated as

$$E = \sum_{i=1}^{n} [X(i)]^2 \quad (2)$$

Where $i$ refers to the number of current vectorframe,X*(I)* is the current frame of speech and $n$ the total number of frames used to calculate energy $E$. The minimum and maximum energies of the word calculated using average of the first few frames are used to cut down the undesired parts of the signal.

### 2.1.5 Dynamic Time Warping (DTW)

DTW finds the best mapping with the minimum distance using Dynamic Programming (DP). The method is called "time warping" since both *x* and *y* are usually vectors of time series and we need to compress or expand in time to find the best mapping.

Let *t* and *r* be two vectors of lengths m and n, respectively. The goal of DTW is to find a mapping path {(*p1, q1*), (*p2, q2*)…(*pk,qk*)} such that the distance on this mapping path is minimized.

The minimum distance is found using DP, which can be summarized in the following three steps:

Step 1: Optimum-value function - Define *D(i, j)* as the DTW distance between *t(1:i)* and *r(1:j)*, with the mapping path starting from *(1,1)* to *(i, ,j).*

Step 2: Recursion:

$$D(i,j) = |t(i) - r(j)| + \min\{D(i-1,j), D(i-1,j-1), D(i,j-1)\} \quad (3)$$

with the initial condition *D(1,1)=|t(1) −r(1)|*.

Step 3: The final answer is D*(m,n).*

### 2.1.6 Feature Extraction

This process involves dividing each word signal into windows of frames and transformation of these frames into a sequence of acoustic feature vectors [9].

#### 2.1.6.1 Windowing

A speech signal is not stationary with its properties changing in time. This limitation is overcome by extracting features from a small window of coefficients where speech is assumed to be stationary. This design uses a Hamming window function for this windowing process. The function is given by Eq. (4).

$$W(n) = \begin{cases} 0.54 - 0.46\frac{\cos(2\pi n)}{L}, & for\ 0 \le n \le L-1 \\ 0 & otherwise \end{cases} \quad (4)$$

where $L$ is the length of the window and $n$ is instance of time.

#### 2.1.6.2 Feature Vector Transformation

The process involves implementing MFCC based feature extraction model. The Cepstrum is used to improve phone recognition performance. The Cepstrum (C) of a signal is the Inverse Discrete Fourier Transform (IDFT) of the log magnitude of Discrete Fourier Transform (DFT) of a signal and is given by Eq. (5).

$$C[k] = \sum_{m=0}^{N-1} \log\left(\left|x[k]e^{-j\frac{2\pi kn}{N}}\right|\right)e^{j\frac{2\pi kn}{N}} \quad (5)$$

where $0 \le k \le N-1$ *and* $0 \le n \le N-1$,$N$ is the total length of the feature vector and $j$ is used for representing a complex number.

#### 2.1.6.3 MFCC with 12 coefficients

Human ear is less sensitive to high frequencies. To avoid it frequencies above the DFT range are mapped to the mel scale. The mel frequency Coefficients (12 in this design) (*mel*) can be computed from the raw acoustic frequency (*f)* using the Eq. (6).

$$mel(f) = 2595 * \log_{10}(1 + \frac{f}{700}) \quad (6)$$

Conversion using mel scale involves creating a bank of triangular filters which collect energy from each frequency band with 10 filters spaced linearly below 1000 Hz and remaining filters spread logarithmically above 1000Hz. By computing the cepstrum the MFCC coefficients are obtained.

### 2.1.7. Hidden Markov Models (HMM)

In HMM based modeling, the feature vectors are the observations and a spoken word usually refers to the states. This study uses the left-to-right model. Each speech sample is broken into several sub samples with equally spaced intervals depending upon the number of states using the parameters extracted and modeled using HMMs. The model is then updated during training and compared with similar models of real time speech signal, during word recognition. An HMM ($\lambda$) is defined using a compact notation given in Eq. (7).

$$\lambda = (\pi, A, B) \quad (7)$$

Where $\pi = \{\pi_i\}$ the probability of being in state $i$=1 initially at time $t$=1, $A$ refers to the state transition probability from state $i$ to state $j$ and is given by Eq. (8).

$$A = \{a_{ij}\} \text{ and } a_{ij} = P(i_{t+1} = j | i_t = i) \quad (8)$$

Where $a_{ij}$ refers to the probability transition to state $j$ at $t$+1 from state $i$ at time $t$ and $B$ is the probability of observing a symbol $v_k$ at state j and is given by Eq. (9).

$$B = \{b_k\} \text{ where } b_k = P(v_k \text{ at } t | i_t = i) \;\; i = 1 \text{ to } N \text{ and } K = 1 \text{ to } M \quad (9)$$

where $N$ represents the number of states and $M$ is the total number of observation symbols. The usage of HMM in modeling real-time applications is by solving three problems namely training, decoding and evaluation. In this study these three parameters $\pi, A$ and $B$ are calculated initially from the feature vector to create the HMM model with 5 states using K-means clustering algorithm [11].

### 2.1.8. Parameter Estimation of HMM model

The parameters of reference HMMs are estimated using a simplified Expectation-Maximization algorithm [5]. A description of the algorithm is given below. The trained HMM is stored in memory as a reference model.

#### 2.1.8.1 Expectation Maximization Algorithm (EM)

The EM algorithm is predominantly used for re-estimating the mean, co-variance and mixture coefficients and updating the HMM parameters to able the reference word models for robust recognition.

It consists of the following steps.

Step 1: The expectation process is performed to calculate the underlying variables used for creating an HMM model. The variables are mixture coefficients $C$, mean $\mu_i$ and covariance $s_i$.

Step 2: It is the maximization process where new estimates of the mean, mixture density and covariance are used to update the observation pdf of the current HMM.

Step 3: The change in the current iteration is used to compare with the previous iteration and a stopping criterion is calculated. The stopping criterion is compared with a threshold value and the iteration is stopped if the stopping criterion is achieved.

Step 4: The iterations are continued until the stopping criteria is achieved.

### 2.1.8.2 Multiple Instance Training (MIT)

When only one observation sequence is used for training, the accuracy of the model is limited. In this study words from 5 speakers are used for training. The single-instance trained model has constraints on the maximum number of HMM states and generates limited number of observations per state. There are *k* observation sequences as shown below Eq. (10).

$$o = \{o^{(1)}, o^{(2)}, ...., o^{(k)}\} \quad (10)$$

Where $o(k) = \{o_1^{(1)}, o_2^{(2)}....,o_{Tk}^{(k)}\}$ is the *k*th observation sequence.

If every observation sequence is independent of one another, the parameters of the model are estimated to optimize the model parameters. The re-estimation formula is given by Eq. (11).

$$P(o|\lambda)\prod_{k=1}^{k} = P(o^{(k)}|\lambda) = \prod_{k=1}^{k} P_k \qquad (11)$$

The re-estimation formulas for multiple observations are obtained by modifying the addition of individual frequencies. The modified re-estimation equations (12), (13) are given below

$$a'_{ij} = \frac{\sum_{k=1}^{k}\frac{1}{P_k}\sum_{t=1}^{T_k-1}[\alpha_t^k(i)a_{ij}b_j(o_{t+1}^{(k)})\beta_{t+1}^k(j)]}{\sum_{k=1}^{k}\frac{1}{P_k}\sum_{t=1}^{T_k-1}[\alpha_t^k(i)\beta_t^k(i)]} \quad (12)$$

$$b'_j(l) = \frac{\sum_{k=1}^{k}\frac{1}{P_k}\sum_{t=1}^{T_k-1} s.t\, o_{t=v_t}[\alpha_t^k(i)\beta_t^k(i)]}{\sum_{k=1}^{k}\frac{1}{P_k}\sum_{t=1}^{T_k-1}[\alpha_t^k(i)\beta_t^k(i)]} \quad (13)$$

where $a'_{ij}$ is the state transition probability, $b'_j(l)$is the observation probability, α is the forward variable, $\beta$ is the backward variable and $l$ is the number of observation.In this method for each sequence will appear in each of the sum over $t$ and disappear in the $P_k$ term and hence will get cancelled exactly.

### 2.2. Testing Phase

The different stages in the testing phase of a speech recognition system is given in Figure 2.5.

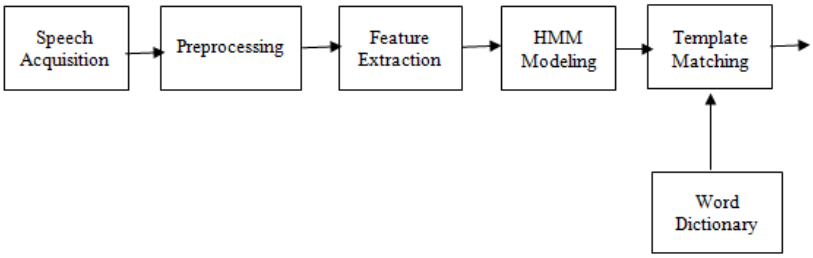


Figure 2.5. Testing Phase of Speech Recognition

The feature vectors of the sentence to be tested are transformed with the same techniques adopted in the training phase [3]. The likelihood of observations calculated using the forward algorithm which uses the formula given in Eq. (14).

$$\alpha_{t+1}(j) = \sum_{i=1}^{N} \alpha_t(i)\, a_{ij}\, b_j(o_{t+1}) \qquad (14)$$

where $\alpha_t(i)$ is the forward variable, $a_{ij}$ is the state transition probability and $b_j$ is the probability of the observation in state j. For each word the probability of observations given the model is calculated. The model with the maximum likelihood is recognized as the spoken sentence using Eq. (15).

$$v = \underset{1<v^*<V}{argmax}[P(O|\lambda^{v^*})] \qquad (15)$$

where $v^*$ is the recognized word, $P(O|\lambda^v)$ is the probability of observation given the model and *V* is the total number of words [3].

# 3. Results and Analysis

### 3.1 Experiment 1:

The training data set consists of about 45 words that are part of 11 sentences commonly associated with smart devices are chosen. The Word Error Rate (WER) of the stand-alone HMM system is estimated in Table 3.1. Word error rate (WER) is a common metric of the performance of a speech recognition or machine translation system.

$$WER = \frac{S + D + I}{N} \qquad (16)$$

where

S is the number of substitutions,

D is the number of deletions,

I is the number of insertions and

N is the number of words in the reference

Table 3.1. WER of three HMM based systems

| No | Sentences Used | HMM | HMM+DTW | HMM+DTW+MIT |
|---|---|---|---|---|
| 1 | Good Morning Praveen. | 27.3 | 26.9 | 27.1 |
| 2 | This is Samsung S6. | 26.1 | 25.6 | 25.8 |
| 3 | This device has Touch Screen | 20.8 | 21.1 | 21.7 |
| 4 | There are two cameras | 19.4 | 20.3 | 17.2 |
| 5 | It has android operating system | 24.5 | 23.3 | 25.4 |
| 6 | It supports up to 64 GB memory card | 26.1 | 24.8 | 24.5 |
| 7 | It supports Bluetooth | 24.1 | 25.2 | 21.5 |
| 8 | It has a long range Wi-Fi | 24.6 | 25.5 | 24.1 |
| 9 | There are free office and pdf editors | 25.1 | 24.9 | 23.9 |
| 10 | The Wi-Fi is not turned on | 23.9 | 22.1 | 24.3 |
| 11 | It has video and photo editing software | 23.3 | 20.3 | 21.3 |
| Average WER | | 24.11 | 23.64 | 23.35 |

The WER of the HMM system was obtained by taking the average of each sentence and was estimated to be 23.4.

### 3.2 Experiment 2:

The WER of the stand-alone is high as experiment was conducted under natural environment where there is noise interference. To improve the accuracy, the signals were preprocessed using DTW and the WER was estimated. Finally, the errors in pronunciation were reduced by performing MIT. The results of the experiments are given in Tables 3.1 and 3.2 and results are compared with a column chart in Figure 3.1.

The decrease in WER percentage between two systems is also analysed using Eq. (17)

$$\%Decrease = \left| \frac{WER_{S2} - WER_{S1}}{WER_{S1}} \right| \times 100 \qquad (17)$$

Table 3.2: Performance of Three Speech Recognition Systems

| No | Technique | WER (Rounded up) |
|---|---|---|
| 1 | Stand-alone HMM | 24.1 |
| 2 | DTW+HMM | 23.6 |
| 3 | DTW+HMM+MIT | 23.4 |

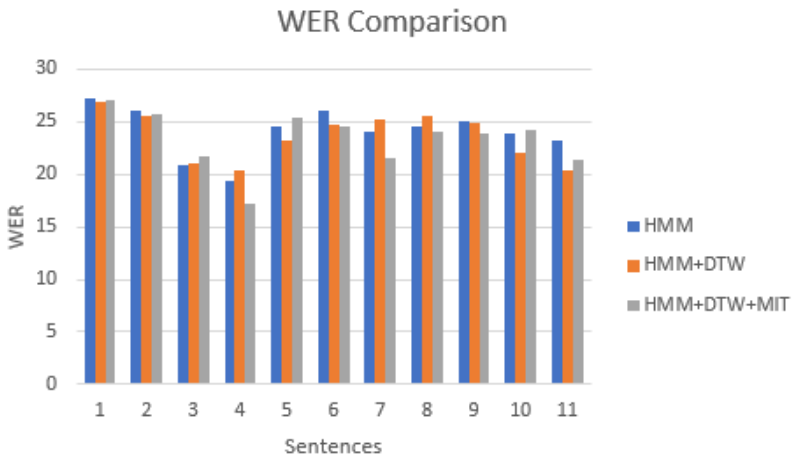


Figure 3.1 WER comparison chart

The continuous word recognition system is based on HMM due to the requirement of simplicity in design. Its performance is better than existing single-instance recognition systems. When the number of instances is increased, training time also increases but is insignificant due to the target application chosen as there is a limit on the vocabulary size (50 words). It is shown in Table. 3.2 that the WER of the proposed continuous speech recognition system is 23.4. It is low when compared to the existing HMM system due to the processing techniques used but, high when compared to the alternative techniques.

The results reveal that the HMM model is the state of the art in modeling speech or phonetic signals but, there is degradation in performance when predicting HMM states. From the results we can calculate the percentage decrease in WER between the different systems. The WER of the HMM+DTW system decreases by 2.07% after preprocessing with DTW. Also, the WER of the HMM+DTW+MIT decreases from the basic system (HMM) by 2.9% after preprocessing with DTW and MIT training.

# 4. Conclusion

In this study, it has been observed that the proposed speech recognition system is stable and reliable and hence suited for resource constrained environments. This system is better in performance when compared with the existing systems for a vocabulary of 50 words. The vocabulary of words must accommodate all possible combinations of words used for smart device control and words that represent complicated settings. Variants of the system already exist with limited performance and techniques like MIT and DTW enables the system to perform better in natural environment.

# References


[1] S. Husnjak,D. Perakovic, I. Jovovic,Possibilities of using speech recognition systems of smart terminal devices in traffic environment. Procedia Engineering, 69, 2014, 778-787.

[2] L. R. Rabiner. A tutorial on Hidden Markov Models and selected applications in speech recognition,Proceedingsof the IEEE, 77(2), 1989, 257-286.

[3] G.Hinton,L.Deng, D. Yu, G.E. Dahl, A.R. Mohamed, N. Jaitly andB. Kingsbury, Deep Neural Networks for acoustic modeling in speech recognition: The shared views of four research groups. IEEE Signal Processing Magazine, 29(6),2012,82-97.

[4] P. A. Srichai,Implementation of a connected digit recognizer using continuous Hidden Markov Modeling, 1998.

[5] T.Hoang, V. V. Quoc, andT. N. L.Thien, FPGA architecture of HMM-based decoder module in speech recognizer. In Control, Automation and Information Sciences (ICCAIS), 2012 International Conference on IEEE, Nov 2012, 354-358.

[6] A. W.Tan, M. V. C Rao, B.Sagar, DA discriminative signal subspace speech classifier. IEEE Signal Processing Letters, 14(2),2007,133-136.

[7] M., Gales and S. Young,The application of Hidden Markov Models in speech recognition. Foundations and trends in signal processing, 1(3), 2008, 195-304.

[8] V. Z. Këpuska and H. A. Elharati, Robust Speech Recognition System Using Conventional and Hybrid Features of MFCC, LPCC, PLP, RASTA-PLP and Hidden Markov Model Classifier in Noisy Conditions. Journal of Computer and Communications, 3(06), 2015, 1.

[9] K.Miura, H. Noguchi, H. Kawaguchi, and M. Yoshimoto, A low memory bandwidth Gaussian Mixture Model (GMM) processor for 20,000-word real-time speech recognition FPGA system. In ICECE Technology, 2008, FPT 2008. International Conference on IEEE FPT, Dec 2008, 341-344.

[10] G.He, Y. Miyamoto, K. Matsuda, S. Izumi, H. Kawaguchi, and M. Yoshimoto, A 54-mw 3×-real-time 60-kword continuous speech recognition processor VLSI. IEICE Electronics Express, 11(2), 2014,20130787-20130787.