HMM BASED SPEECH RECOGNITION OF CONTINUOUS THAI DIGITS
Montri Karnjanadecha, Patimakorn Kimsawad and Pongtorn Tunthanakit

Department of Computer Engineering
Faculty of Engineering
Prince of Songkla University
Hatayai, Songkla, THAILAND 90112
Phone:+66-74-212-895, Fax:+ 66-74 -212-895
Email: montri@coe.psu.ac.th

ABSTRACT
Progress on speech recognition of Thai digit strings is presented in this paper. HTK 3.0 was chosen to implement the HMM-based speech recognizer. MFCCs and their delta and delta-delta terms were used as speech features. Several set of HMM parameters were investigated. Two kinds of word searching methods were tried. Recognition accuracy of 98.7% on test data was achieved with a fixed length word network when ignoring silence in performance calculation.

1. INTRODUCTION
There has been a number of research on Thai speech recognition. However, most of the works concerned the recognition of Thai digit in isolated-word mode using various techniques including Dynamic Time Warping (DTW) [1], Artificial Neural Networks (ANNs) [2] and Hidden Markov Models (HMMs) [2]. Highest accuracy reported on isolated-word Thai digit recognition was 99.1% from a neural network classifier with Perceptual Linear Prediction (PLP) and tone modeling as front end analysis [3].

Automatic digit recognition systems can be applied to many applications, for example: in telephone dialing systems and numerical data entry systems. Although there are only 10 digits to recognize, a major difficulty lies on that the pronunciation of a digit or a digit string is not restricted to a grammar like that of the conversational speech. Thus language models cannot be used to further improve recognition accuracy. As a consequence, system performance has to rely entirely on speech characteristics. For conversational speech, human can guess the missing words or mis-recognized words from the context of the spoken sentences. In contrast, for digit, there is no contextual information for a listener to take advantage of. An applicable automatic digit recognizer must be a system that provides 100% accuracy. This is a challenging goal.

This work reports a progress in the development of a continuous Thai digit recognition. We believe that a major factor that has inhibited the progress of continuous speech recognition in Thailand was the lack of speech recognition tools. Most of speech researchers in the country have developed their own tools to fulfil their needs. The problem was that it is not easy to build a workable tool to support the recognition of continuous speech. It requires a huge amount of knowledge, time and efforts. This has been a reason why most of the published research works were concerned isolated word recognition.

The objective of this work is to build an automatic speech recognizer that can accurately recognize strings of Thai digits. Hidden Markov Model Toolkit (HTK) version 3.0 [4] was chosen to be our tools.

HTK has been known to be a powerful set of speech recognition tools and be in use in many institutions around the world. This software is now freely downloadable form the Internet1. HTK provides several types of feature extraction methods, including Linear Predictive Coefficients (LPCs) and Mel-Frequency Coefficients (MFCs). In addition, the variations of these feature types (e.g. LPCCs, MFCCs, etc.) can be computed. HTK supports discrete and continuous density HMMs.

In this work, a set of continuous density HMMs were built and trained to recognize strings of Thai digits. Several combinations of HMM model parameters and two types of searching methods were tried. Highest recognition accuracy on test data of 98.7% was achieved with use of a fixed 7-digit search network.

This paper is organized as follows. Section 2 discusses background. Section 3 discusses methods. Experiments, discussion of results, and conclusions are presented in Section 4, Section 5 and Section 6, respectively.

2. BACKGROUND
Thai digit recognition is one of the tasks in speech recognition areas. It has been studied for over 10 years. In the beginning, the task began with an easy one, which was the recognition of Thai digits pronounced in isolation. At that time a simple template matching technique was investigated. A well-known technique called DTW was widely used to compare the similarity of

1 http://htk.eng.cam.ac.uk
utterances so that a decision can be made. DTW employs the Dynamic Programming (DP) technique to non-linearly align two utterances. Its goal is to minimize the distance (usually Euclidean) between them. Although DTW has been shown to be promising, especially for speaker dependent speech recognition, it has a main drawback in that it requires multiple templates for each sound in order to accurately recognize the speech in speaker independent fashion. Recognition time is greatly increase with such approach and it seems to limit itself to only a system with small vocabulary size and speaker dependent. There were attempts in applying the DTW on Thai digit recognition, for example, work in [5].

Next solution for isolated-word Thai digit recognition has been to use ANNs (usually multi-layer perceptron with Backpropagation training) for speech modeling [3][6][7]. ANN is a powerful pattern classification framework that has been successfully applied in many areas. Since speech utterances are of various lengths, thought they might be of the same word and pronounced by the same speaker, the feature extraction part of the system has to be able to parameterize each utterance into a fixed-length feature vector. This feature vector is then fed to the neural network whose number of inputs is also fixed.

Currently, the most powerful and popular pattern classifier for speech recognition applications is HMM. HMM is a state machine with two independent stochastic processes. It can take observation sequence of any length as input. In other words, utterances of various length can be nicely taken care of by the HMM mechanism. Besides, the HMM framework can be extended to handle continuous speech effectively and efficiently without major modification in training or recognition algorithm. Practically, one HMM is used to model one sound unit. A sound unit could be a phone, a syllable, or a word. HMM parameter estimation can be efficiently achieved using Baum-Welch algorithm. Viterbi algorithm can be used in recognition phase to find the best model that matches the test utterance.

There has been only a few numbers of research works that applied HMM on Thai speech recognition. One of the works was presented in [8].

3. METHODS

The purpose of this work was to build a HMM-based, speaker-independent, continuous Thai digit recognizer that is able to accurately recognize unknown digit strings. HTK Version 3.0 was entirely used in this work. Every part of the system is described below.

3.1 Feature Extraction

Waveform of each digit string was first pre-emphasized with a high-pass filter whose transfer function was $H(z) = 1-0.97z^{-1}$. Then the filtered waveform was divided into frames. Each frame was 25 ms long and was Hamming windowed. Spacing between frames was 10 ms.

For each frame, 12 Mel-Frequency Cepstral Coefficients (MFCCs) plus an energy term were computed. The MFCC terms were calculated from bins of 24 filters spreading in Mel scale across the frequency range of 70-8,000 Hz. The energy term represented the log-scaled energy of the signal in the frame.

The combination of 12 MFCCs and an energy term resulted in a total of 13 parameters per frame. These were then augmented with their delta and delta-delta coefficients. The resulting number of parameters per frame became 39. Note that the delta terms were computed over a span of 5 frames of the MFCC parameters and the delta-delta terms were computed over a span of 5 adjacent delta terms.

Speech parameters mentioned above were computed using HTK’s HCopy tool.

3.2 HMM Parameters

There were 11 different sounds to model in this work (10 digits + silence), thus 11 HMMs were required. Note that the entire realization of each digit is modeled with one HMM. This was a whole word-based, not a phone-based recognition.

The HMMs used in this work were continuous density type with diagonal covariance matrices. Each HMM had 39 inputs. The number of states was varied from 4-6 and the number of mixtures per state was varied from 1-3. Only self-transition and transition to the next state were allowed.

Each HMM model was initialized with Viterbi training for 20 iterations (using HTK’s HInit tool), then trained with the Baum-Welch algorithm for 20 iterations in isolated word mode (using HTK’s HRest tool) and for 3 iterations in embedded mode (using HTK’s HRest tool).

3.3 Recognition

Recognition was accomplished using Viterbi recognition algorithm (using HTK’s HVite tool). An important information needed to provide the HVite tool to perform the recognition is word lattice or word network. The word network tells the recognizer how words are related (or connected). It specifies all possible sets of word sequences. Use of an appropriate word network can greatly improve recognition accuracy and speed because non-existing word sequences will be ignored.

In this work, two kinds of word networks were investigated. The first network allows multiple occurrences of digits. This can be illustrated as a
graph in Fig. 3.1.

![Fig. 3.1 Word network for multiple occurrences of digits.](image)

The Start node and the End node indicate the entry point and exit point, respectively. A valid word sequence can be determined by moving along the arrow direction. The loop-back link implies that each of the 11 sounds can occur more than once. Thus recognized digit sequences can range from 1 digit long to infinity.

The second network depicted in Fig. 3.2 limits the length of each string to 7 digits. Between a pair of digits, there is silence as an option, where it may or may not present. For this network, the recognized string will always be 7 digits long (excluding silence). Note that digit denotes an occurrence of a digit from the set of 0 to 9.

![Fig 3.2 Word network for strings of 7 digits.](image)

3.4 Performance Evaluation

In the recognition step, the most likely transcription of each test utterance was obtained. The recognized results were compared against the reference transcriptions. Then the recognition accuracy was computed. These steps were achieved using HTK’s HResults tool. All the performance figures reported in this paper are in % accuracy, in which insertion, deletion and substitution errors have been accounted for.

4. EXPERIMENTS

All experiments presented in this paper were conducted in speaker independent mode. The number of speakers used was 22 (18 males + 4 females). Each speaker was prompted to utter 10 digit strings, each with 7 randomly generated digits. There were a total of 220 digit strings, which were then manually labeled with time indices. Although there were no silences in the prompts, speech utterances with silence between digits were allowed and were labeled as ‘sil’.

Speech data were recorded in an office environment with the air condition turned on. Recording equipments consisted of a personal computer, a commercial grade soundcard and a condenser microphone. Speech was digitized with 16-bit resolution at 16,000 samples per second.

Since the number of speaker was rather small and we wanted to keep as many of training speakers as possible so that the speaker independent speech models could be reliably estimated, we managed to rotate the speakers in training and test set. First, all utterances from 20 speakers were used as training data and recognition was performed on the remaining utterances, which were now served as test data. Then utterances from other two speakers in the training set were taken out and the previous test utterances were place back into the training set. The steps were repeated (11 iterations) until all utterances have been used as test data. The recognized strings were accumulated and the recognition accuracy was computed. This training scenario was used in all experiments.

The first experiment was conducted using the front end analysis and HMM parameters as described in Section 3. The recognition was performed using the word network in Fig. 3.1. Performance evaluation was done in two difference approaches. The first one was to immediately compare the recognized transcriptions against the reference transcriptions. In this case the recognition of silences were also taken into consideration. The second approach was to remove all occurrences of silences in both reference transcriptions and

---

2 In conversational speech, silence can be used to
recognized transcriptions prior to the comparison. In this case, accuracy on the recognition of silences did not affect the accuracy figure. Table 4.1 depicts the results of this experiment.

Table 4.1  % accuracy on test data at various numbers of states and mixtures, with and without silence in performance evaluation, using word network in Fig. 3.1.

<table>
<thead>
<tr>
<th>#States</th>
<th>#Mix.</th>
<th>w/ Silence</th>
<th>wo/ Silence</th>
</tr>
</thead>
<tbody>
<tr>
<td>4</td>
<td>1</td>
<td>64.6</td>
<td>75.0</td>
</tr>
<tr>
<td>4</td>
<td>2</td>
<td>73.1</td>
<td>89.0</td>
</tr>
<tr>
<td>4</td>
<td>3</td>
<td>70.6</td>
<td>88.5</td>
</tr>
<tr>
<td>5</td>
<td>1</td>
<td>74.9</td>
<td>90.7</td>
</tr>
<tr>
<td>5</td>
<td>2</td>
<td>78.0</td>
<td>95.1</td>
</tr>
<tr>
<td>5</td>
<td>3</td>
<td>76.5</td>
<td>95.6</td>
</tr>
<tr>
<td>6</td>
<td>1</td>
<td>77.9</td>
<td>94.9</td>
</tr>
<tr>
<td>6</td>
<td>2</td>
<td>78.9</td>
<td>96.7</td>
</tr>
<tr>
<td>6</td>
<td>3</td>
<td>78.5</td>
<td>96.7</td>
</tr>
</tbody>
</table>

The second experiment was similar to the first one except that the word network in figure 3.2 was used in the recognition phase. The results are summarized in Table 4.2.

Table 4.2  % accuracy on test data at various numbers of states and mixtures, with and without silence in performance evaluation, using word network in Fig. 3.2.

<table>
<thead>
<tr>
<th>#States</th>
<th>#Mix.</th>
<th>w/ Silence</th>
<th>wo/ Silence</th>
</tr>
</thead>
<tbody>
<tr>
<td>4</td>
<td>1</td>
<td>82.0</td>
<td>96.8</td>
</tr>
<tr>
<td>4</td>
<td>2</td>
<td>82.5</td>
<td>98.0</td>
</tr>
<tr>
<td>4</td>
<td>3</td>
<td>82.2</td>
<td>98.3</td>
</tr>
<tr>
<td>5</td>
<td>1</td>
<td>83.5</td>
<td>98.0</td>
</tr>
<tr>
<td>5</td>
<td>2</td>
<td>83.3</td>
<td>98.7</td>
</tr>
<tr>
<td>5</td>
<td>3</td>
<td>82.8</td>
<td>98.6</td>
</tr>
<tr>
<td>6</td>
<td>1</td>
<td>83.9</td>
<td>98.4</td>
</tr>
<tr>
<td>6</td>
<td>2</td>
<td>83.2</td>
<td>98.5</td>
</tr>
<tr>
<td>6</td>
<td>3</td>
<td>83.1</td>
<td>98.2</td>
</tr>
</tbody>
</table>

5. DISCUSSION OF RESULTS

The results in Table 4.1 show that ignoring silence in accuracy calculation can greatly reduce the recognition error. Best recognition accuracy obtained was 78.9% when silences were intact and the accuracy was as high as 96.7% when all silences were ignored. This resulted in almost 7 times in error reduction.

Use of a strict word network such as the network in Fig. 3.2 resulted in a much higher accuracy than that in the previous experiment. As depicted in Table 4.2, the highest accuracy was 83.9% and 98.7% when silences were intact and ignored, respectively.

Obviously, 5- and 6-state HMMs performed equally well and use of 2 mixtures per state was a good compromise.

6. CONCLUSIONS

This paper presents a progress in the recognition of continuously spoken Thai digits, using a HMM based recognizer. Recognition accuracy on test data of 98.7% was achieved due to the use of a fixed length word network. Such word network could be utilized in any application that requires digit string of fixed length, for example in telephone dialing systems. Ignoring of silences in the performance evaluation phase can also greatly improve recognition accuracy. This work can serve well as a benchmark for the task. Even though the accuracy is perceivably high, it is still far from our 100% accuracy goal. This implies further investigations in this area.

REFERENCES

[8] Areepongs, S. and Jitapunkul, S., “Speaker...