Monosyllabic Thai Tone Recognition Using Ant-Miner Algorithm

Saritchai Predawan[†], Chom Kimpan[††], and Chai Wutiwiwatchai[†††]

[†], [††] Faculty of Information Technology Rangsit University, Muang-Ake, Pathumthani, 12000 Thanland
[†††]National Electronics and Computer Technology Center, Ministry of Science and Technology,
112 Thailand Science Park, Klong Luang, Pathumthani 12120 Thailand

Summary
Recognition of tone is essential for speech recognition and language understanding. A monosyllabic Thai tone recognition system, which is based on the Ant-Miner algorithm. The system is composed of three main process, fundamental frequency (F0) extraction from input speech signal, analysis of F0 contour for feature extraction, In the F0 feature extraction, the polynomial regression functions are employed to fit the segmented F0 curve where its coefficients are used as a feature vector[8]. In tone recognition, we used the Ant-Miner classifier to classify a tone by assuming that the feature is features vector. The hypothetical words used in this paper are composed of numerical words and monosyllabic Thai words. The experimental results show that by using the system as a speaker-dependent system, the maximum recognition rate is 96.20%. These results indicate that the intrinsic structure of tone can be exploited to reduce the need for costly labeled training data for tone learning and recognition.

Key words:
Thai Tone Recognition, Ant-Miner Algorithm, Feature Extraction, Fundamental Frequency (F0)

1. Introduction

Tone and intonation play a crucial role across many languages. However, the use and structure of tone varies widely, ranging from lexical tone which determines word identity to pitch accent signaling information status. There are five tones in Thai language, low("ake"), medium("saman"), falling("toe"), high("dtee") and rising("jattawa"). The feature of speech that was used to classify the tone is the shape of fundamental frequency (F0) contour, which shown in figure 1. There are several parameters that also have the effect on the shape of F0 contour such as the gender and the age of speaker, the initial consonant, the final consonant and the duration of vowel (short or long). Intonations in Thai are used to distinguish the word's meanings[1]. Research in Thai tone extraction presented by Ramalingam [4] is compared with our work. Besides, several Thai speech recognition approaches that have been elaborated for years such as in [2] and [5] are studied. The correctness values of these approaches are 74.9% [2] and 89.4% [5]. In our process, the F0 contour of input speech was automatically smoothed and segmented by the proposed algorithm in section 3. Then they were fit by the polynomial regression function, which we used its coefficients as the features of F0 contours. In the recognition process, we used Ant-Miner algorithm.

This paper is organized as follows. First, the overview of the speech recognition system is introduced. Second, the phonetics of Thai language. Third, F0 feature extraction, Ant Colony optimization and the proposed system of this experiment. Fourth, the experimental result is explained. The conclusion and discussion is given in the section 5.

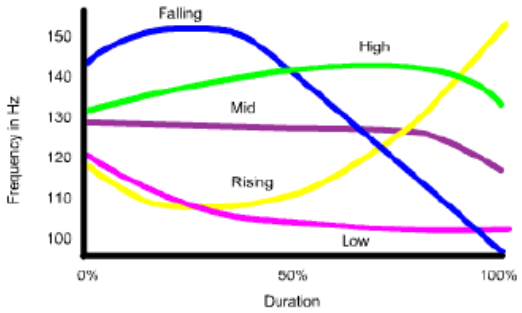

Figure 1 F0 normalize contour pattern of five Thai Tones

2. Thai Syllable Structure

The Thai syllable structure is composed of three different sound systems as follows [3].

1. Theoretically, the Thai language has 69 letters, which can be grouped into three classes of phone expression: consonant, vowel and tone. There are 44, 21, and 4 letters for consonants, vowels, and tones, respectively. Some Thai consonant symbols share the same phonetic sound. Because of this, there are only 21 phones for Thai consonants. On the other hand, some vowels can be combined with other vowels, resulting in 32 possible phones. However, in practice, only 18 letters in the vowel class are currently used in Thai. There are 4

Manuscript received January 5, 2009
Manuscript revised January 20, 2009

tone symbols to express 5 Thai tones. Thus, there are 66 practical letters as shown in Table 1.

Table 1: Three classes of Thai alphabet: consonants, vowels and tones.

Basic Classes	Letters in each class
Consonant (44)	ก,ข,ฃ,ค,ฅ,ฆ,ง,จ,ฉ,ช,ซ,ฌ,ญ,ฎ,ฏ,ฐ,ฑ,ฒ,ณ,ด,ต,ถ,ท,ธ,น,บ,ป,ผ,ฝ,พ,ฟ,ภ,ม,ย,ร,ล,ว,ศ,ษ,ส,ห,ฬ,อ,ฮ
Vowel (18)	อั, อะ, อี, อา, อิ, อี, อึ, อื, อุ, อู, เอ, แอ, ไอ, อำ, ไอ, ไอ, ฤ, อ์
Tone (4)	อ่, อ้, อี, อ๋

2. Like most languages, a Thai syllable can be separated into three parts; (1) initial consonant, (2) vowel and (3) final consonant. The phonetic representation of one syllable can be expressed in the form of $/C_i\text{-}V^T\text{-}C_f/$, where C_i is an initial consonant, *V*is a vowel, C_f is a final consonant and *T* is a tone which is phonetically attached to the vowel part. Some initial consonants are cluster consonants. Each of them has a phone similar to that of a corresponding base consonant. For example, *pr*, and *pl* are similar to a base consonant *p*. In the vowel part, there are 18 vowel phones and 6 diphthongs. Following the concept presented in [6], there are totally 76 phonetic symbols and 5 tone symbols in Thai, as shown in Table 2. Naturally phones, especially those in the vowel class, are various in their durations. In Thai language, most vowels have their pairs. For example, the vowel pair *a* and *aa* have a similar phone but different durations. The other vowel pairs are *i-ii, v-vv, u-uu, e-ee, x-xx, o-oo, @-@@, q-qq, ia-iia, va-vva,* and *ua-uua*. Intuitively, these pairs are easily confused in the recognition process.

Table 2: Phonetic symbols grouped into initial consonants, vowels, final consonants and tones.

Initial Consonant (C_i)		Vowel (*V*)	Final Consonant (C_f)	Tone (*T*)
Base	Cluster			
p,t,c,k,z,ph, th,ch,k,h,b, br,bl,d,dr, m,n,ng,r,f, fr,fl,s,h,w,j	*pr,phr,pl, phl,tr,thr, kr,khr,kl, khl,kw, khw*	*a,aa,i,ii,v,vv,u, uu,e,ee,x,xx,o, oo,@,@@,q, qq,ia,iia,va, vva, ua, uua*	*p^,t^,k^,n^,m^,n^, g^,j^,w^,f^,l^,s^, ch^, jf^,ts^*	0 (Mid) 1 (Low) 2 (Falling) 3 (High) 4 (Rising)

3. Thai alphabets of different classes have different styles of pronunciation. The consonantal letters can be uttered in either of the following two styles. The first style is simply pronouncing the core sound of a consonant. For example, the letter 'ก', its core sound can be represented as the phonetic sound */k-@@0/*. Normally, some consonants share a same core sound. For example, 'ค' ,'ค', 'ฆ' have the same phonetic sound */kh-@@0/*. In such case, the hearer may encounter with letter ambiguity. To solve this issue, the second style is generally applied by uttering a core sound of the consonant followed by the representative word of that consonant. Every consonant has its representative word. For example, the representative word of the letter 'ก' is "ไก" (meaning: "chicken", sound: */k-a1-j^/*), and that of the letter 'ข' is "ไข" (meaning: "egg", sound: */kh-a1-j^/*). To express the letter 'ก' using this style, the syllable sequence */k-@@0/+/k-a1-j^/* is uttered.

Expressing letters in the vowel class is quite different from that of the consonant class. There are two types of vowels. The first-type vowels can be pronounced in two ways. One is to pronounce the word "สระ" (meaning: "vowel", sound: */s-a1//r-a1/*), followed by the core sound of the vowel. The other is to simply pronounce the core sound of the vowel. On the other hand, for the second-type vowels, they are uttered by calling their names. The vowel letters of each type are listed in Table 3. As the last class, tone symbols are always pronounced by calling their names. Table 4 concludes how to pronounce a letter in each alphabet class.

Table 3: Two types of vowels

First-type vowels	อะ, อา, อิ, อี, อึ, อื, อุ, อู, เอ, แอ, ไอ, อำ, ไอ, ไอ
Second-type vowels	อั, อี๋, อ์, ฤ

Table 4: Pronouncing methods for each alphabet class

Alphabet class	Pronouncing methods
Consonant	1. consonant core sound + representative word of consonant
	2. consonant core sound
First-type vowel	1. */s-a1//r-a1/* + vowel core sound
	2. vowel core sound
Second-type vowel	1. the vowel name
Tone	1. the tone name

The smallest construction of sounds or syllables in Thai is composed of one vowel unit or one diphthong, one two or three consonants, and a tone. The construction can be represented with the structure as illustrated in Figure 2,

$$\mathbf{S = C_i(C_i)V^T(V)(C_f)}$$

Figure 2. Thai Syllable Structure

Where C_i is initial consonant, C_f is final consonant, V is vowel, and T is tone respectively. In our study we divide syllable into phonemes based on the time that each phoneme occurs and their characteristics. These phonemes are: 1) the initial consonant,
2) the vowel,
3) the secondary vowel,
4) the syllable ending,
5) the tonal. Considering the following examples.

/a:/ has two phonemes, tonal (medium) and vowel (a:) phonemes.

/ma:/ has three phonemes, tonal (medium), initial consonant (m), and vowel (a:).

/ku:a:m/ has five phonemes, tonal (medium), initial consonant (k), vowel (u:), secondary vowel (a:), and syllable ending (m).

From the above example, it can be summarized that the tonal is the most important phoneme embedding in every Thai syllable. Our research also investigated that the tone always locates on the vowels and the resonant syllable ending phonemes. The following example demonstrates the characteristic of Thai language that syllables have their meaning base on phonemes' pitch contouring.

/ka:L/ with low tone means a kind of plant.
/ka:M/ with medium tone means a kind of grass
/ka:F/ with falling tone means to kill.
/ka:H/ with high tone means to trade.
/ka:R/ with rising tone means a leg.

3. Analysis of Thai tones

From section II, we concluded that the tonal phonemes always place on the vowel phoneme. A block diagram of the proposed system based is depicted in Figure 3. The proposed system consisted of four modules: preprocessing, syllable segmentation, feature extraction, and tone classifier.

In the preprocessing module, shown in Figure 4, a speech signal was first low-pass filtered and then blocked into frames. The pitch detection and energy computation were performed. The syllable segmentation module located the endpoints of the spoken syllables by utilizing the relationships between the peaks and valleys in the modified energy contour.

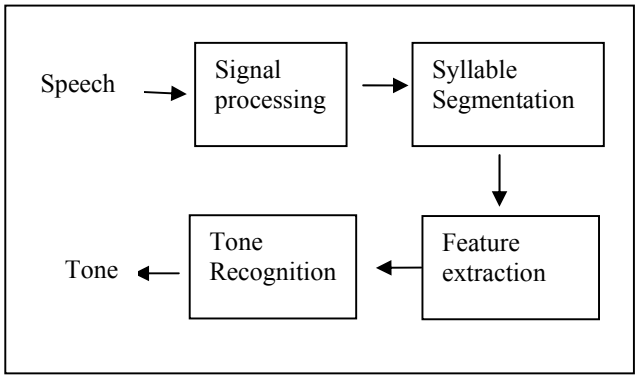

Figure 3 Block Diagram of The System

The statistical data analysis of the acoustical features [9]; including duration, energy, and fundamental frequency (F0) of syllables are computed from the training sets uttered by all speakers. It was found that duration and normalized energy are the effective features for distinguishing between the stressed and unstressed syllables, while the mean normalized F0 did not signal the stress function of the syllables. Intonation affects the tone patterns by making them decline gradually [11, 12].

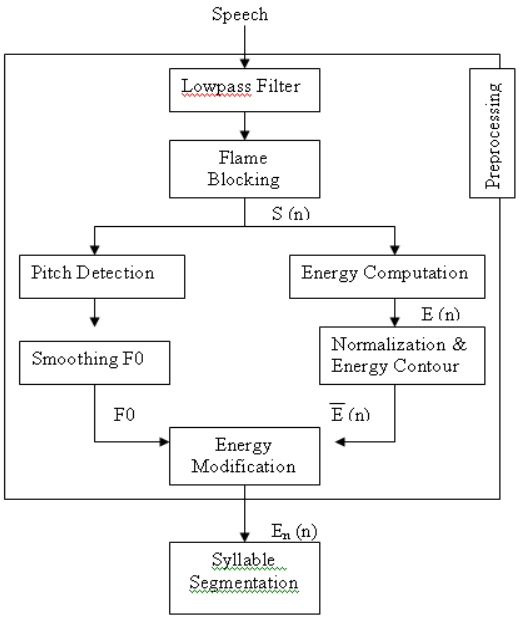

Figure 4 Processing Module

Within each tone, the mean F0 of the preceding syllable is higher than the succeeding syllable, and the mean F0 is lowest at the ending syllable of the sentence. This datum suggests that the mean F0 can he used to deal with the intonation effect. The tone pattern of a syllable is also affected by tone patterns of the neighboring syllables due to the anticipatory and carryover co-articulation. It is evident that F0 contours of both stressed and unstressed syllables are subject to modification by the preceding and succeeding syllables.

3.1 F0 Feature Extraction

The F0 feature extraction process has two procedures. The first is F0 smoothing and segmentation procedure.

(I) F0 Smoothing And Segmentation Procedure : F0 from the F0 extraction process will be smoothed in the smoothing procedure by using median filtering. In the segmentation procedure, there is algorithm that was used to segment the smoothed F0. This algorithm will determine the beginning and the ending frame of the longest time that F0 at each frame has the value differ from the neighboring frame no more than Δ Fmax =17 Hz.

(II) Polynomial Regression : The objective of this procedure is to determine the coefficients β of a polynomial that fits the segmented F0 contour.

Let :

$F = (F1, F2,...,FL)^T$ be a sequence of segmented

F0 of length L ,

F^ (F^1, F^2,...,F^L)^T be an estimated vector of F

A d-dimension feature vector β = (β0, β1,..., βd −1)T is the coefficient of (d-1)-order polynomial regression function

$$\hat{F}_i = \beta_0 + \beta_1 t_i + \beta_2 t_i^2 + \ldots + \beta_{d-1} t_i^{d-1} \quad (1)$$

Where t_i = i/L is a normalized time respect to F_i . Equation (1) can be expressed in matrix form as (2).

$$\hat{\mathbf{F}} = \mathbf{T}\boldsymbol{\beta}, \quad \begin{bmatrix} \hat{F}_1 \\ \hat{F}_2 \\ \vdots \\ \hat{F}_L \end{bmatrix} = \begin{bmatrix} 1 & t_1 & t_1^2 & \cdots & t_1^{d-1} \\ 1 & t_2 & t_2^2 & \cdots & t_2^{d-1} \\ \vdots & \vdots & \vdots & & \vdots \\ 1 & t_L & t_L^2 & \cdots & t_L^{d-1} \end{bmatrix} \begin{bmatrix} \beta_0 \\ \beta_1 \\ \vdots \\ \beta_{d-1} \end{bmatrix} \quad (2)$$

A solution for in a least-squares sense (minimize the Euclidean distance between vectors F and F) is obtained via forming the pseudo-inverse of T, that is (3)

$$\boldsymbol{\beta} = (T^T T)^{-1} T^T F \quad (3)$$

However, if T^TT is nearly singular, the numerical errors incurred in forming T^TT , and then forming the inverse, spawn a need for alternate approaches that are not plagued by numerical sensitivities

3.2 Classification Algorithms

(I) Ant Colony Optimization : The ACO algorithm [6] is an essential system based on some agents that simulate the natural behaviors of ants. The natural behaviors are include mechanisms of cooperation and adaptation (Natural behavior of ants is shown in Figure 5). It is based on the following ideas.

- Each path is followed by an ant which is associated with a candidate solution for a given problem.
- When an ant follows a path, the amount of pheromone deposited on that path is proportional to the quality of the corresponding candidate solution for the target problem.
- When an ant has to choose between two or more paths, the first priority path is the path, which has a larger amount of pheromone.

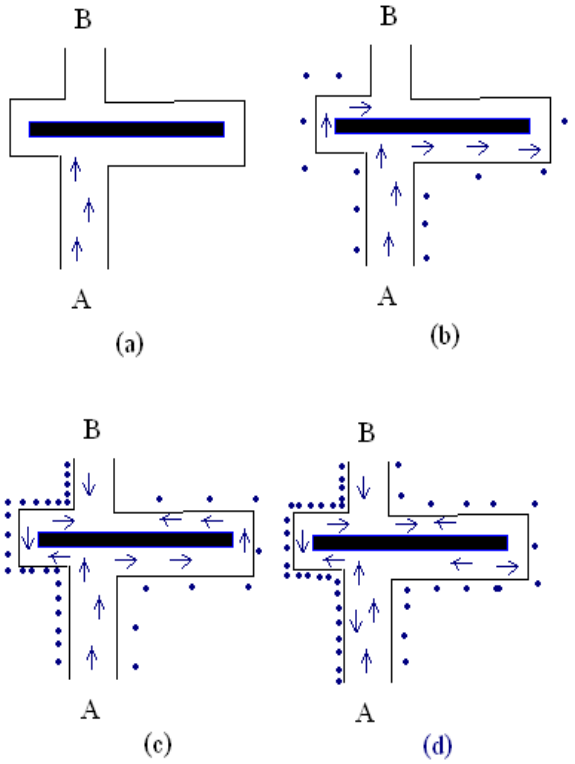

Figure 5 How real ants find a shortest path from A to B. (a) Ants arrive at a decision point. (b) Some ants choose the left path and others choose the right path. The choice is random. (c) The ants which choose the shorter path (left) arrive at the opposite decision point faster than longer path (right). (d) Pheromone accumulates at a higher rate on the shorter path. The number of points is approximately proportional to the amount of pheromone deposited by ants.

The ACO is used to solve various kinds of trace route and congestion problems. As a result, the ant eventually converges on the shorts path (optimum or a nearest-optimum solution).

(II) Ant-Miner algorithm : The Ant-Miner algorithm [7,10] has been proposed to the set of IF-THEN rule in the form of IF < term1 and term2 and … > and THEN < Class > in the work of data mining. Each term in the rule of antecedent part is a triple < attributes, operator, value >. Value is the possible value in a domain of each attribute. There is only an operator "=" used in this work such as < Month = January >.

```
TrainingSet = {all training cases};
DiscoveredRuleList = [ ]; /* initialize rule list with empty list */
WHILE (TrainingSet > Max_uncovered_cases)
    t = 1; /* ant index, and also rule index */
    j = 1; /* convergence test index */
    Initialize all trails with the same amount of pheromone;
    REPEAT
        Ant_t starts with an empty rule and incrementally
        constructs a classification rule R_t by adding one
        term at a time to the current rule;
        Prune rule R_t;  /* remove irrelevant terms from rule */
        Update the pheromone of all trails by increasing
        pheromone in the trail followed by Ant_t (proportional
        to the quality of R_t) and decreasing pheromone in the
        other trails (simulating pheromone evaporation);
        IF (R_t is equal to R_t – 1) /* update convergence test */
            THEN j = j + 1;
            ELSE j = 1;
        END IF
        t = t + 1;
    UNTIL (i ≥ No_of_ants) OR (j ≥ No_rules_converg)
    Choose the best rule R_best among all rules R_t constructed by
    all the ants;
    Add rule R_best to DiscoveredRuleList;
    TrainingSet = TrainingSet - {set of cases correctly covered
    by R_best};
END WHILE
```

Figure 6 High level of the Ant-Miner algorithm

The consequent part specifies the class prediction in case that predicted attributes satisfy all the terms in the antecedent part. The set of rules, constructed by this algorithm cover all or almost all the training cases. These rules have a small number of terms that is good for data mining. The high level of its algorithm is shown in Figure 6.

Following the algorithm; after the pheromone is initialized, many rules are constructed in the inner Repeat-Until loop with the rule pruning and the pheromone updating method. The loop will stop when ants construct the same rule continually more than *No_Rule_Converg* times or the number of rules is equal to the number of ants. When the inner Repeat-Until loop is completed, the best rule will be added to the rule list. Then, all training cases predicted by this rule were removed from the training case set. Pheromone is initialized again. This cycle is controlled by the Outer Repeat-Until loop.The Repeat-Until loop will be finished when the number of uncovered training cases is less than a threshold, called *Max_uncovered_cases*. The rule discovered in the Repeat-Until loop that has the highest quality is then added to the list of discovered rules, and the training examples correctly covered by that rule are removed from the training dataset. An example is correctly covered by a rule if the example satisfies the rule antecedent and has the class predicted by the rule.

(III) Pheromone Initialization : Pheromone values for each term are all initialized to the same value at the beginning of each While loop iteration. The initial value of each pheromone is given by the function (4) :

$$\tau_{ij}(t=0) = \frac{1}{\sum_{i=1}^{a} b_i} \tag{4}$$

Where a is the total number of attributes, i is the index of an attribute, j is the index of a value in the domain of attribute i, and bi is the number of values in the domain of attribute i.

(IV) Pheromone Updating : In Ant-Miner pheromone levels are increased for all terms in a rule just constructed by an ant, based on the quality of that rule, as measured by the rule quality formula (5) “sensitivity * specificity”, defined as follows:

$$Q = \frac{TP}{TP+FN} \cdot \frac{TN}{FP+TN} \tag{5}$$

Where TP / (TP + FN) is the sensitivity, TN / (FP + TN) is the specificity, and:

TP (true positives) is the number of cases covered by the rule that have the class predicted by the rule.

FP (false positives) is the number of cases covered by the rule that have a class different from the class predicted by the rule. FN (false negatives) is the number of cases that are not covered by the rule but that have the class predicted by the rule. TN (true negatives) is the number of cases that are not covered by the rule and that do not have the class predicted by the rule.

(V) Term Selection: The probability that a term will be added to the current rule is given by the following formula (6) :

$$P_{ij} = \frac{\eta_{ij} \cdot \tau_{ij}(t)}{\sum_{i=1}^{a} x_i \cdot \sum_{j=1}^{b_i} (\eta_{ij} \cdot \tau_{ij}(t))} \tag{6}$$

Where: η_{ij} is the value of a problem-dependent heuristic function – more precisely information gain [6] – for $term_{ij}$ (a condition of the form attribute i = value j). The higher the value of η_{ij} the more relevant for classification the

$term_{ij}$ is, and so the higher its probability of being chosen. $\tau_{ij}(t)$ is the amount of pheromone associated with $term_{ij}$ at iteration t. a is the total number of attributes.

4. EXPERIMENT AND RESULT OF RECOGNITION

To evaluate the performance of the proposed speech recognition system, the speech material used in the experiment was a Thai isolated words database produced by 26 speakers (13 males, 13 females, 780 wave files), within the range of 20-35 years old. The speech utterances were recorded in a quite room. The recorded speech is 8-bits and 16 kHz sampling rate. The utterances from 20 speakers (10 males, 10 females, 600 wave files) were used as training data, and 180 wave files were use as test set. Both experiments use 30 monosyllabic words as shown in Table 5.

Table 5 List of 30 monosyllabic words for Tone Classification

Tone	Mid (M)	Low (L)	Falling (F)	High (H)	Rising (R)
Words	/'dqqn0/	/'paak1/	/'wing2/	/'nok3/	/'huu4/
	/'n@@n0/	/'pet1/	/'kluuaj2/	/'to3/	/'svva4/
	/'taa0/	/'kaj1/	/'som2/	/'nam3/	/'s@@ng4/
	/'mvv0/	/'hnvng1/	/'nang2/		/'saam4/
	/'thiian0/	/'sii1/	/'kxxw2/		/'suun4/
	/'kin0/	/'hok1/	/'haa2/		
	/'tiiang0/	/'cet1/	/'kaw2/		
		/'pxxt1/			

The resulting raw F0 contours were smoothed using median filtering and linear interpolation. Syllable onset and offset were determined from a simultaneous display of a wide-band spectrogram, energy contour, F0 contour and audio waveform using conventional rules for segmentation of the speech signal. The tone classification step is based on Ant-Miner Algorithm method. The first step is to identify the possible tone sequences in the extracted F0 contour. The resulting contours are then compared to the smoothed and normalized input F0 contour in the processing module. To avoid generating all possible tone sequences to match against a test sentence, peak-and-valley analysis is used to reduce the number of reference templates. Given the smoothed, normalized and segmented F0 contour for a test sentence, local extreme (peaks and valleys) are detected by using first and second derivatives. The derivative at any point in the contour, except for the first two and last two points, is computed by calculating the linear regression coefficients of a group of five F0 values consisting of the current point, and its preceding and following two points.

The same procedures are applied in the training and recognition stages.

(I) Data : There are 30 Thai monosyllabic wave files from 26 speakers. The total utterances are 780 wave files.

(II) Pre-processing : In this step. A single syllable is its input. a speech signal was first low-pass filtered and then blocked into frames. The pitch detection and energy computation were performed. The syllable segmentation module located the endpoints of the spoken syllables by utilizing the relationships between the peaks and valleys in the modified energy contour. F0 was computed in this process from 256 samples speech frame with the overlapping of ¾ frames by using modified short-term autocorrelation with center clipping method.

(III) Feature Extraction : The F0 feature extraction process, the features of each utterance are extracted which determines the parameters that have sufficient information to describe the shape of F0 contour by the method of polynomial regression. An example, which features extracted of "1" utterance has initial consonant /HN/, vowel /V/, final consonant /NG/ and low tone, each frame consists of 256 sample data.

(IV) Recognition Process : In the this step,"1" which has 13 attributes for each data are classified by the Ant-Miner algorithm. The rule list from the algorithm is used as the recognition engine.

4.1 Recognition Engine and Recognition Rule

In this experiment, the Ant-Miner algorithm is used for training the recognition system (to construct a rule list). The utilizing data was made from feature of a speech utterance described in section 3.The original version of the Ant-Miner [7,10], the quality of the rule is computed by the equation (7), the rules from the algorithm is short. It has a minimum number of rules in the rules list with a high accuracy. (Cover all or almost all in the training set). In this experiment, recognition all of the training data is needed. The equation of Q (Rule) is changed to (4)

$$Q(Rule) = \left(\frac{TP}{FP+1} \right) \tag{7}$$

By the Q (Rule), the value of FP is converged to zero. This means that accuracy rate when testing, is converged to most data. The number of rules will be more than the original. The other parameters are No_of_Ant, No_Rule_Converg and Max_Uncovered_Cased. In the training step, the data of 600 wave files into 5 classes (all of monosyllabic words), which each class has 39 attributes for each data. Finally data of each group are classified by the Ant-Miner algorithm. It was described in section 4. The rule list from the algorithm is used as the recognition engine.

4.2 Karhunen-Loeve Transformation

An orthogonal transform called the Karhunen-Loeve Transform (KLT) has been well known in principal components analysis or hotelling transformation. It has

been applied in various applications in which input data are enormous such as image processing, speech processing etc. The main purpose is to decompose the signals into completely decorrelated components in the form of empirical basis functions that contain the majority of the variations in the original data. The decorrelated components are often called eigenvectors and the scaling constants used to reconstruct the spectra are generally known as coefficient vectors, as shown in Figure 7.

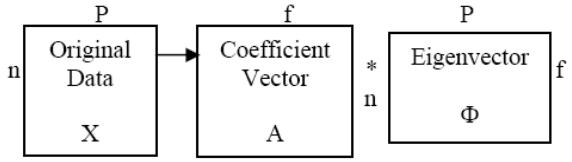

Figure 7 The KLT breaks apart the spectral data into the most common spectral variations (eigenvectors) and the corresponding coefficient vectors.

The Karhunen-Loeve decomposition process is started by adapting the original matrix to be a square matrix. The covariance matrix ($\sum_x$) calculation, shown in (8), is selected to achieve in this approach. Given original data

$X = [x(t_1)\ldots x(t_n)]^T$, the covariance matrix of X is ($\sum_x$) where

$$(\sum_x)_{jk} = E[\{x(j) - \overline{x(j)}\}\{x^*(k)-\overline{x^*(k)}\}] \quad (8)$$

where P is the total number of input data vector. Let Φ_i, i = 1,…,n, be the eigenvectors of ($\sum_x$), calculated form (9) for which their associated eigenvalues λ_i are arranged in descending order such that $\lambda_1 \geq \lambda_2 \geq \ldots \geq \lambda_n$.

$$S\Phi_i = \lambda_i \Phi I \quad (9)$$

Those terms associated with smaller eigenvalues can be discarded. In this approach, we choose the eigenvectors projected from eigenvalues that have the total equal 95% of the whole summation of eigenvalues as word representative. The eigenvector Φ_i has a coefficient vector A_i associates with it, computed using (10)

$$A_i = X\Phi_i^T \quad (10)$$

The coefficient vectors, of dimension P, are the projection of the original data matrix X used to monitor the changes over time for each dominant eigenvector. The transformation matrix used to reconstruct the original image is formed the *f* eigenvectors corresponding to the largest *k* eigenvalues. The X′ vectors will be *f*–dimensional and the reconstruction is given by (11)

$$X' = A_K\Phi + \overline{m} \quad (11)$$

It can be shown in the following section that the mean square error MSE between X′ and X for a specified *k* the error is minimized by selecting the *k* eigenvectors associated with the largest eigenvalues. There are two ways to investigate the properly amount of principal components to represent the whole set of data with lossless information. The first is plot graph between calculates eigenvalue and their amplitudes as shown in Figure 8.

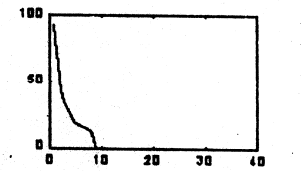

Figure 8 eigenvalue rank vector for word '0'

Form Figure 8, the higher magnitude of eigenvalue are falled between 0 – 10, therefore we can discard all components with zero-values magnitudes without loss information. The another way to selected the number of principal components is calculated by summing all eigenvalue and approximately 90-95% of it is quite good for representation. In our approach, we select 95% of total sum.

4.3 Learning Vector Quantization (LVQ)

The Learning Vector Quantization (LVQ) is a self-organization network, originated in the neural network community by Kohonen introduction [12] as shown in Figure 9. The LVQ is a method for non-parametric classification that uses a "winner-take-all" strategy. Learning vector Quantization classifies its input data into classes that it determines by using various learning algorithms. The codebook vectors are created from a selected group of input vectors and then used as initial vectors. The trained and tested vectors are created from the other groups of input vectors and then used in training and classifying processes, respectively.

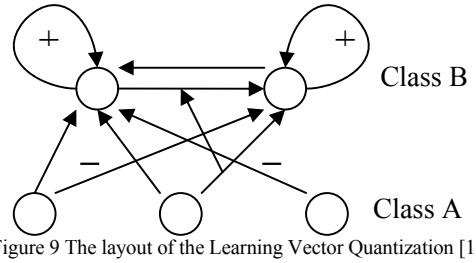

Figure 9 The layout of the Learning Vector Quantization [13]

The training set of 600 wave files are input in the Ant-Miner algorithm for learning and classify each tone class of 30 hypothetical words by construction of a set of rules. The result is show in Table 6, and averages are calculated in each tone class.

Table 6 Recognition Rate of Proposed System.

Desired Tone	Recognized Tone					Total	Accuracy (%)
	M	L	F	H	R		
Mid (M)	36	6	0	0	0	42	85.71
Low (L)	0	48	0	0	0	48	100.00
Falling (F)	2	0	40	0	0	42	95.24
High (H)	0	0	0	18	0	18	100.00
Rising (R)	0	0	0	0	30	30	100.00
						180	96.20

The results of comparing Ant-Miner and KLT+LVQ are reported in Table 7. KLT+LVQ was run with the default settings for its parameters. To make the comparison as fair as possible, both algorithms used exactly the same training and test set partitions in each of the iterations of the recognition procedure.

Table 7 Recognition rate of proposed system and KLT+LVQ[9]

Recognition Engine	Recognition Rate
Proposed System	96%
KLT+LVQ[9]	90%

5. CONCLUSION

We have demonstrated the effectiveness of ant-miner algorithm for monosyllabic Thai tone recognition. We used the coefficient of polynomial regression function as a feature vector of the segmented F0 contour. In the training phase, the feature vector was used to determine the statistical parameters of the model for each class of tone. In the testing phase, the feature vector will be passed to the automatic recognition process. The result shows that the system can recognized 96.20% of the testing set. The recognition rate of the propose system is more than the KLT+LVQ [9]. There are many advantages of the proposed system. First, where there are some feature parts which are incorrect or missing, it will use the other appropriate features. It is the solution for the problem [7, 10].The second is giving a higher recognition rate. We believe that this method can be applied to other types of recognition.

References

[1] R.Kongkachandra, S.Pansong, T.Sripramong and C.Kimpan. : "Thai Intonation Analysis in Harmonic-Frequency Domain," The 1998 IEEE APCCAS Proceeding, (1998) pp. 165-168. 1998

[2] E.Maneenoi, et al. : "Modification of BP Algorithm for Thai Speech Recognition," NLPRS'97 (Incorporating SNLP'97) Proceeding (1997), pp. 287-291.

[3] Luksaneeyanawin, S. : "Intonation in Thai," Ph.D. Thesis, University of Edinburgh, 1983.

[4] Ramalingam, H. : "Extraction of Tones of Speech : An Application to The Thai Language," Master Thesis (TC-95-5), Asian Institure of Technology, Thailand, 1995.

[5] W.Pornsukjantra, et al: "Speaker Independent Thai Numeral Speech Recognition Using LPC and the Back Propagation Neural Network," NLPRS'97 (Incorporating SNLP'97) Proceeding, pp. 585-588. 1997.

[6] Marco Dorigo, Thomas Stutzle. "Ant colony optimization," A Bradford Book The MIT Press, Cambridge, Massachusetts, London, England, 2004.

[7] S. Airphaiboon. "Recognition of Hand-written Thai character considering the head of character," Masters Thesis, Department of Electrical Engineering, King Monkut's Institute of Technology Ladkraband, Bangkok, Thailand, 1998.

[8] P. Charnvivit, S. Jitapunkul et al., "F0 Feature Extraction by Polynomial Regression Function for Monosyllabic Thai Tone Recognition," Eurospeech 2001, Scandinavia.

[9] S. Predawan, P. Jiyapanichku. C, Kimpan and C. Wutiwiwatchai. "Tone Analysis in Harmonic-Frequency Domain and Reduction using KLT+LVQ for Thai Isolated Word Recognition," 2006 WSEAS International Conference, May 2006.

[10] P.Phokharatkul, K.Sankhuangaw, S.Phaiboon S.Somkuarnpanit, and C.Kimpan "Off-Line Hand Written Thai Character Recognition Using Ant-Miner Algorithm," Transactions on ENFORMATIKA on Systems Sciences and Engineering, vol. 8, pp. 276-281, October 2005.

[11] A. Deemagarn and A. Kawtrakul. "Thai Connected Digit Speech Recognition Using Hidden Markov Models," SPECOM'2004 : 9th Conferenece Speech and Computer, September 2004.

[12] A.Biem, S.Katagiri and B.-H.Juang, "Discriminative feature extraction for speech recognition," International workshop on Neural Networks for signal proceesing, Baltimore September 1993.

[13] M.Purat, T.Liebchen and P.Noll. "Lossless Transform Coding of Audio Signals," 102nd AES Convention, Muchen, Preprint No.4414, 1997.

Acknowledgments

First of all, I thank you my family, for listening and supporting in the last couple of years. I deeply thank my advisor, Associate Professor Dr. Chom Kimpan, whose help, advice and supervision was invaluable. I also thank you Dr.Chai Wutiwiwatchai, whose advice and support the data for this research.

Saritchai Predawan received B.Sc in Computer Science, Faculty of Science, Burapha University, Thailand, and his M.Sc in Science Education (Computer), Faculty of Graduate Studies, King Mongkut's Institute of Technology Ladkrabang, Bangkok, Thailand. His interests include artificial intelligence, swarm intelligence, pattern recognition and ant colony optimization.

Chom Kimpan received D.Eng. in Electrical andComputer Engineering from King Mongku1t'sInstitute of Technology Ladkrabang, M.Sc. in Electrical Engineering from Nihon University, Japan, and B.Eng. in Electrical Engineering from King Mongkut's Institute of Technology Ladkrabang, Bangkok, Thailand. Now he is anassociate professor at the Department of Informatics, Faculty of Information Technology, Rangsit University, Thailand. His interests are in pattern recognition, image retrieval, speech recognition and swarm intelligence.

Chai Wutiwiwatchai received the Doctor Degree of Speech Technology, Department of Computer Science, Tokyo Institute of Technology, Japan, M.Eng (Digital Signal Processing, Electrical Engineering) from Chulalongkorn university in 1998 and his B.Eng (Electrical Engineering with the 1st honor) from Thammasat University in 1994. He joined the Software and Language Engineering Laboratory (SLL), National Electronics and Computer Technology Center (NECTEC) as an Assistant Researcher since July 1998. His expertise is on speech processing especially speech and speaker recognition,which are currently researched at NECTEC. His interest is on pattern recognition and speech technology.