International Journal of Information, Control and Computer Sciences
ISSN: 2517-9942
Vol:1, No:8, 2007

Off-Line Hand Written Thai Character Recognition using Ant-Miner Algorithm

P. Phokharatkul, K. Sankhuangaw, S. Somkuarnpanit, S. Phaiboon, and C. Kimpan

***Abstract*—**Much research into handwritten Thai character recognition have been proposed, such as comparing heads of characters, Fuzzy logic and structure trees, etc. This paper presents a system of handwritten Thai character recognition, which is based on the Ant-minor algorithm (data mining based on Ant colony optimization). Zoning is initially used to determine each character. Then three distinct features (also called attributes) of each character in each zone are extracted. The attributes are Head zone, End point, and Feature code. All attributes are used for construct the classification rules by an Ant-miner algorithm in order to classify 112 Thai characters. For this experiment, the Ant-miner algorithm is adapted, with a small change to increase the recognition rate. The result of this experiment is a 97% recognition rate of the training set (11200 characters) and 82.7% recognition rate of unseen data test (22400 characters).

***Keywords*—**Hand written, Thai character recognition, Ant-miner algorithm, distinct feature.

I. INTRODUCTION

THE goal of "handwritten Thai character recognition" is to make a computer understand and specify which Thai character a human wrote. This research can be applied to many different types of information. For example transformations of handwriting into text files for any purpose of MIS application. This research proposes 2 major character recognition methods. The first one is description of the three features in each character image (Head, Endpoint and Feature code) [1, 2, 3] The second one is the Ant-Miner algorithm [4, 5, 6].

Previous research used the three features to construct the recognition rules as the "Structure tree". The head and the position of characters are the main consideration point; and then the other features are considered again. The disadvantage of the structure tree is wrong character recognition: when characters are written incorrectly. This may occur when the head is in the wrong position, is incomplete, is missing, or when the head is larger than it should be. In this research, we used 112 characters (76 alphabetic and 36 symbol characters) per one handwritten sheet image as a data test. The handwritten characters are collected from 100 persons. Each person made 3 copies of a sheet. The total sample of characters is 33600. After the head features of all collected characters are extracted, it was found that 36% of all the collected characters' heads were incorrect. The recognition rate would be less if the method of a structure tree is used. In this research, the Ant-miner algorithm is used for recognition rule construction. Though there are some parts which are incorrect or missing, the algorithm will use the other appropriated features. It is the solution for the problems [1, 3] and also gives a higher recognition rate.

II. THAI CHARACTERS

Thai alphabets consist of 76 characters and many special symbols often use in Thai language. Thai characters are composed of circles, lines, curves, and zigzags. Examples of handwritten Thai characters are illustrated in figure 1 and all characters used in this experiment are shown in table 1.

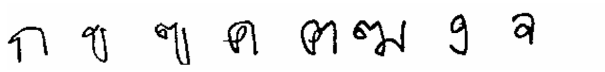

Fig. 1 Examples of handwritten Thai characters

TABLE I
SHOW ALL CHARACTERS USED IN THE EXPERIMENT

<table>
<tr><th>Thai characters</th><th>Others symbols and others characters used in Thai language</th></tr>
<tr><td>๐ ๑ ๒ ๓ ๔ ๕ ๖ ๗ ๘ ๙
ก ข ฃ ค ฅ ฆ ง จ ฉ
ช ซ ฌ ญ ฎ ฏ ฐ ฑ
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 ะ
า เ ไ ใ โ ฤ ฦ อั อ่
อ้ อ๊ อ๋ อิ อี อึ อ็ อํ อ์
อื อุ อู</td><td>- ( ) * , . + _ / \
| : ; ? ‘ ’ " = ฯ ๆ ๏
๛ # $ ^ & ฿ < > [ ]
{ } @ ! %</td></tr>
</table>

P. Phokharatkul is with the Department of Computer Engineering, Faculty of Engineering, Mahidol University, Nakhon Pathom 73170, Thailand (e-mail: egpph@mahidol.ac.th).
K. Sankhuangaw is with the Master of Science Program in Technology of Information System Management, Faculty of graduate studies, Mahidol University, Nakhon Pathom 73170, Thailand (e-mail: xcomscv@hotmail.com)
S. Phaiboonis with the Department of Electronic Engineering, Faculty of Engineering, King Mongkut's Institute of Technology Ladkrabang, Bankok, Thailand (e-mail: egspb@mahidol.ac.th)
S. Somkuarnpanit is with the Department of Electronic Engineering, Faculty of Engineering, King Mongkut's Institute of Technology Ladkrabang, Bankok, Thailand (e-mail: kssuripo@kmitl.ac.th).
C. Kimpan is with the Department of informatics, Faculty of Information Technology, Rangsit University, Pathum Thani, Thailand.

International Journal of Information, Control and Computer Sciences
ISSN: 2517-9942
Vol:1, No:8, 2007

III. Three Features Of Thai Characters with a Pair of Attributes and Their Values

A. *Head of Thai Character*

The head is one of the distinctive features of Thai characters. It is defined as a circle or a closed loop in a character [1]. It is normally a starting point of writing a Thai-language character. Some characters have only one head; some characters have more than one, or even have none, depending on personal habits of writing.

Normally, different characters have different head zones. Thus, the attribute Head_Zone_Z_n is defined by whether the zone z_n has a head or not. The possible value of this attribute is True or False (have or none) as an example character image shows in Fig. 2.

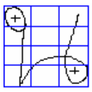

Fig. 2 An example of attribute where Head_Zone_Z_1 = True and Head_Zone_Z_{12} = True but other zones are False

B. *End point*

The end point is the point that has only one point connected to it [1]. Attribute End_point_z_n is used to define it. The possible value of this attribute is True or False whether the zone $\boldsymbol{z_n}$ has an end point or not. For an example, see Fig. 3.

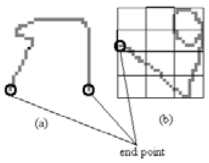

Fig. 3 (a) The position of end point (b) Representing the condition of a term with End_point_Z4 = True

C. *Feature Code*

The feature code is defined by the maximum number of points that the referent lines pass in its zone [1]. Feature_Code_Z_n is used to represent an attribute for this feature. Zone Z_1, Z_2, Z_3 and Z_4 use horizontal referent lines and zone Z_5, Z_6 and Z_7 use vertical referent lines. See figure 4 for an example.

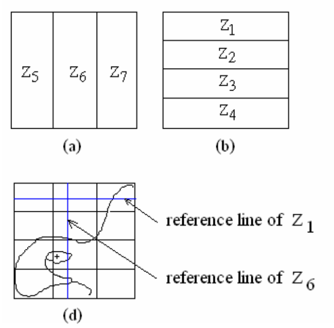

Fig. 4 (a) The zones that use vertical referent lines. (b) The zones that use horizontal referent lines. (d) Two referent lines pass Zones Z_1 and Z_6 that make an attribute to be Feature_Code_Z_1 = 1 and Feature_Code_Z_6 = 5

IV. Ant Colony Optimization

A. *Ant Colony Optimization*

The ACO algorithm [6] is an essential algorithm for this research. This algorithm simulates the natural behavior of ants (The natural behavior of ants is shown in Figure 5). It is based on the following ideas.

1) Each path is followed by an ant which is associated with a candidate solution for a given problem.
2) 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.
3) 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.

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

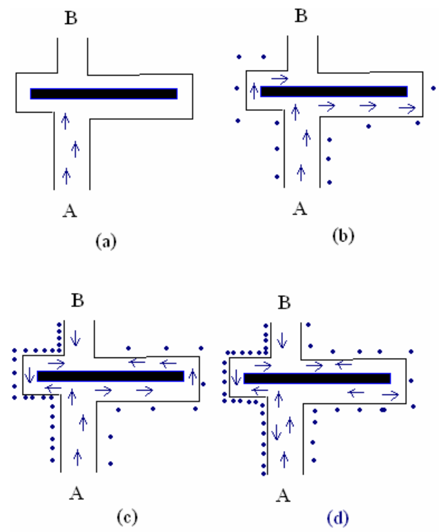

Fig. 5 How real ants find a shortest path from A to B. (a) Ants arrive at a decision point. (b) Some ants choose the left path and others choose the right path. The choice is random. (c) The ants which choose the shorter path (left) arrive at the opposite decision point faster than longer path (right). (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

B. *Ant-Miner Algorithm*

The Ant-Miner algorithm [3] has been proposed to discover a set of IF-THEN rules from data in the form of **IF** < **Term1 AND Term2 AND…** > **THEN** < **Class** > in the work of data mining. Each term in the rule of the antecedent part is a triple attribute, an operator and a 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 >. The consequent part specifies the class prediction only in the case that predicted attributes satisfy all 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.

A small number of rules that is good for data mining. The high level of its algorithm is shown in Fig. 6.

```
Training set = all training cases;
Rule list = empty;
REPEAT
   i=0;
   Pheromone Initialization;
   REPEAT
      i=i+1;
      Ant_i constructs a classification rule;
      Prune the current constructed rule;
      Update the pheromone of the trail followed by Ant_i;
      The best rule is memorized;
   UNTIL (i = No_of_Ants) or (Ant_i constructed the same rule as the
    previous Ants continually No_Rule_Converge times)
   The best rule is added to the rule list;
   Remove the cases covered by the selected rule from the training set;
UNTIL (Number of cases in the Training set less than
Max_uncovered_cases)
```

Fig. 6 High level of the Ant-Miner algorithm

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 which are predicted by this rule are 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 finish when the number of uncovered training cases is less than a threshold, called *Max_uncovered_cases*.

V. IMPLEMENTATION

Overall model of the implementation system is shown in Fig. 7. It consists of 5 steps; convert input data to bitmap, pre-processing, feature extraction, recognition engine, and recognition result.

A. Data

There are Thai hand-written characters from 100 persons. Conditions of writing are free-style writing with magic pen (0.5 diameters). Each person makes 3 copies of a hand-written character sheet with 112 characters per sheet. The total characters are 33600.

B. Convert Data to Bitmap

In this step, data in this experiment is converted to bitmap by a scanner. The result of this step is a two-color bitmap file, 200 dpi.

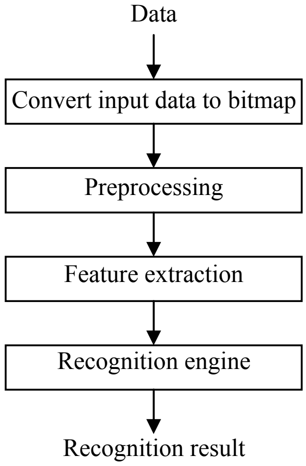

Fig. 7 Overview of system implementation

C. Pre-processing

1. Thinning Method

The character image width is larger than 1 pixel. In the feature extraction process, only a character skeleton is used. The character skeleton is a character image that is only 1 pixel in width (see figure 8 for an example). In this step, the algorithm in the reference [1] is used to convert a normal character into a skeleton image.

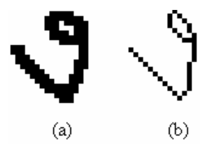

Fig. 8 (a) Normal character image (b) Skeleton character image

2. Zones of Reference

Each character will be normalized to 128x128 pixels and segmented into 12, 9 and 15 zones with the same width and height for feature endpoint, Head and Feature code. The names given to each zone are $\boldsymbol{z_1, z_2, \ldots}$ as shown in figure 9.

z1	z2	z3
z4	z5	z6
z7	z8	z9
z10	z11	z12

Fig. 9 Twelve zone division

3. Feature Extraction

In this step the features of each character in section 3 are extracted and saved to a file for the next step. See figure 10 for an example of the program that extracts features of ‘น’ character.

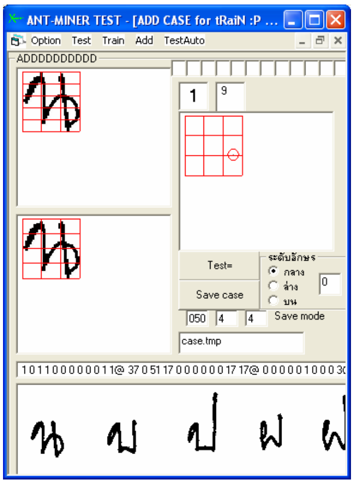

Fig. 10 Software that use to extract the feature of character

D. Recognition Engine and Recognition Rule

1. Algorithm

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 three features of a Thai character described in section V.

In the original version of the Ant-Miner [3], the quality of the rule is computed by the equation (1)

$$Q(Rule) = \frac{TP}{TP+FN} x \frac{TN}{FP+TN} \qquad (1)$$

Where:

a) TruePos (TP) is the number of cases covered by the rule and having the same class as that predicted by the rule.
b) FalsePos (FP) is the number of cases covered by the rule and having a different class from that predicted by the rule.
c) FalseNeg (FN) is the number of cases that are not covered by the rule, while having the class predicted by the rule.
d) TrueNeg (TN) is the number of cases that are not covered by the rule which have a different class from the class predicted by the rule.

With the equation, the rules from the algorithm are 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 training data is needed. Then the equation of Q (Rule) is changed to (2).

$$Q(Rule) = \left(\frac{TP}{FP+1}\right) \qquad (2)$$

By the Q (Rule), the value of FP is converged to zero. This means that accuracy rate when training, is converged to most data. The number of rules will be more than the original. The other parameters are *No_of_ant=1500, No_Rule_Converg= 10 and Max_uncovered_cases = 0.*

2. Training the Recognition Engine

In the training step, the data of 11200 samples from 100 people are classified into 112 classes (all of characters), which has 100 data in each class and 31 attributes for each data. This is done to classify into three levels (Thai language has 3 levels see Figure 11 for details). So, the characters are then classified into 5 groups

a) Upper characters, which are tones and some vowels, eg. อ่ อ้ อิ อ๋ อี อี
b) Upper and middle, which are some vowels and some characters located in-line and above-line eg. ไ ใ โ ฬ ป ฟ ฝ
c) Middle characters, which are the group of characters and vowels located in-line eg. ก ค เ
d) Middle and low characters, which are groups of characters located in-line and under-line eg. ฤ ฎ ฏ ฦ ๆ
e) Lower characters, which are the group of characters located under-line such as อุ อู

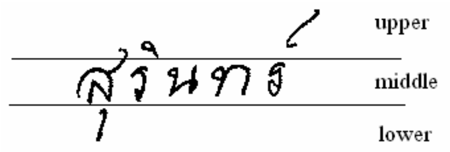

Fig. 11 Three levels of Thai characters

Finally data of each group are classified by the Ant-miner algorithm. It was described in chapter IV. The rule list from the algorithm is used as the recognition engine.

The algorithm, while running to make recognition rules, is shown in Fig. 12. Fig. 13 shows the example of a rule list generated by the Ant-miner algorithm.

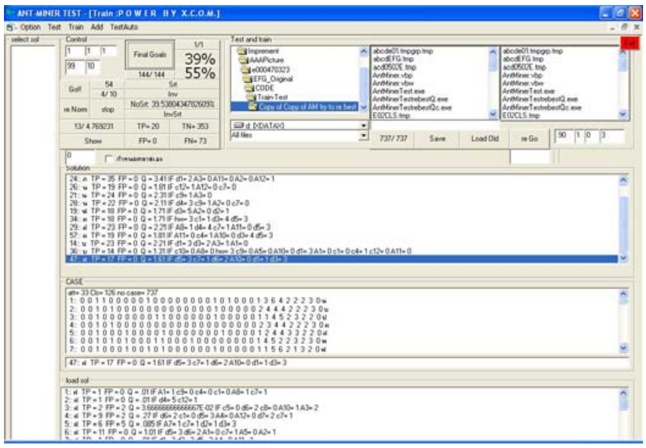

Fig. 12 The algorithm is running to construct the rule list

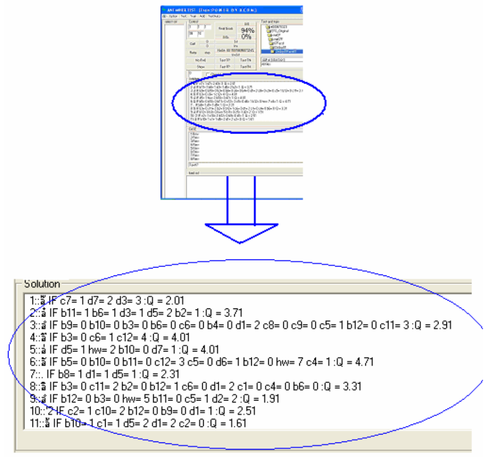

Fig. 13 Example of rule list (a, b and c are three feature end points,Head and feature code. The อ๊, อ๋, อํ, อื, อ่, อิ, and อ้ are classes of characters)

VI. The Experimental Result

The training set of 11200 data are input in the Ant-Miner algorithm for learning. In order to classify each group of handwritten Thai characters by construction of a set of rules. Each group is trained 3 times. Then, the same rule list is used to predict 22400 new data of handwritten Thai character. The result is show in Table II and the best results of the proposed recognition system are shown in Table III. Averages in table III are calculated by weight with numbers of characters in each group. Table IV compares the recognition rate of the proposed system with the Structure tree.

VII. Conclusion

This paper presents a system of handwritten Thai character recognition. It is based on the Ant-Miner algorithm. The result shows that the system can recognize 97% of the training set. In the research, there are many rule lists created by the training process. The recognition rate of the best rule list from various styles of 22400 unseen data is 82.7%. So, the best rule list (Highest recognition rate and Minimum number of rules) is used for recognition-engine construction for Thai character recognition. Because 36% of all the collected characters in this experiment have invalid heads of characters then the recognition rate of the proposed system is more than the structure tree [1, 3]. There are many advantages of the proposed system. First, when there are some feature parts which are incorrect or missing, it will use the other appropriate features. It is the solution for the problems [1, 3]. The second is giving a higher recognition rate. We believe that this method can be applied to other types of recognition. For example Thai printed character recognition and a combination of Thai and English character recognition if using other appropriate features.

TABLE II
Result of Handwritten Thai Character Testing with an Ant-Miner Classifier

Group of characters	Number of characters	Number of rules	Terms per Rule	Recognition rate (%) (Training data)	Recognition rate (%) (Unseen data)
Upper	1700	163	5.2	98.4	88.2
Upper	1700	176	5.4	98.5	87.7
Upper	1700	149	5.5	98.3	87.9
Upper and middle	1900	279	6.1	98.9	85.2
Upper and middle	1900	286	6.0	98.6	83.4
Upper and middle	1900	296	6.3	98.9	82.4
Middle	6300	1587	8.2	92.2	76.5
Middle	6300	1611	8.0	95.5	79.4
Middle	6300	1528	8.4	94.2	77.6
Middle and lower	700	64	4.7	100	86.5
Middle and lower	700	60	4.7	100	85.1
Middle and lower	700	67	4.6	100	86.1
Lower	600	44	3.7	100	89.0
Lower	600	46	3.7	100	88.8
Lower	600	44	3.7	100	90.4

TABLE III
The Best Result of Handwritten Thai Character Testing with an Ant-Miner Classifier

Group of characters	Number of characters	Number of rules	Terms per Rule	Recognition rate (%) (Training data)	Recognition rate (%) (Unseen data)
Upper	1700	163	5.2	98.4	88.2
Upper and middle	1900	279	6.1	98.9	85.2
Middle	6300	1611	8.0	95.5	79.4
Middle and lower	700	64	4.7	100	86.5
Lower	600	44	3.7	100	90.4
Average				97.0	82.7

TABLE IV
RECOGNITION RATE OF PROPOSED SYSTEM
AND STRUCTURE TREE SYSTEM

Recognition Engine	Recognition rate
Proposed system	82.7%
Structure tree [3]	Less than 64%

REFERENCES

[1] P. Choruengwiwat, "Thai handwritten character recognition using extraction of distinctive features," Masters Thesis, Department of Electrical Engineering, Chulalongkorn University, 1998.
[2] T. Thongkamwitoon , W. Asdornwised, S. Aramvith, S. Jitapunkul, "On-line Thai-English handwritten character recognition using distinctive features," APCCAS '02. 2002 Asia-Pacific Conference on Circuits and Systems, vol. 2, p 259 – 264, 2002.
[3] 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 Ladkrabang, Bangkok, Thailand, 1988.
[4] Parpinelli R.S, Lopes H.S, Freitas, A.A., "Data mining with an ant colony optimization algorithm," IEEE Transactions on Evolutionary Computation, p321 - 332, 2002.
[5] Bo Liu, Abbas H.A, McKay B., "Classification rule discovery with ant colony optimization," IEEE/WIC International Conference on Intelligent Agent Technology, p 83 – 88, 2003.
[6] Marco Dorigo, Thomas Stutzle, "Ant colony optimization", A Bradford Book The MIT Press, Cambridge, Massachusetts, London, England, 2004.

Pisit Phokharatkul received the Doctor Degree of Engineering in Electrical and Computer Engineering (2002) from King Mongkut's Institute of Technology Ladkrabang, M.Eng (Electrical Engineering) in 1990 from King Mongkut's Institute of Technology North Bangkok, M.Eng (Nuclear Technology) in 1985 from Chulalongkorn Univerity, and B.Ed in Physics from Sri Nakharinwirot University in 1981, respectively. Now he is an assistant professor of Computer Engineering and Academic Infrastructures Deputy Dean at the Faculty of Engineering, Mahidol University. His interests include pattern recognition, image retrieval, fuzzy logic application, and swarm intelligence.

Katunyoo Sankhuangaw received B.Sc in Physics, Faculty of Science, Chiang Mai University, Thailand. Now, he is studying for a masters degree in Technology of Information System Management, Faculty of Graduate Studies, Mahidol University, Thailand. His interests include artificial intelligence, swarm intelligence, pattern recognition and ant colony optimization.

Suripon Somkuarnpanit received the Ph.D. degree in Optical Electronics and lazer engineering from Heriot-Walt University UK, M.Sc. in Medical Electronics and Physics from Bartholomew's Hospital Medical College, University of London, UK., and B.Eng. in Electrical Engineering from King Mongkut's Institute of Technology Ladkrabang, Thailand. Now he is an associate professor in Electronics at Faculty of Engineering, King Mongkut's Institute of Technology, Ladkrabang. His interests include opto electronics, lazer and lazer applications, pattern recognition, image retrieval and medical instrumentation. He is a member of the IEEE.

Supachai Phaiboon received B. Eng. and M. Eng degrees in Electrical Engineering from King Mongkut's Institute of Technology North Bangkok, Bangkok, Thailand in 1987 and 1990, respectively. He is an associate professor in Electrical at Faculty of Electrical Engineering, Mahidol University. His research interests include radio propagation and measurement, computational intelligence, pattern recognition, speech processing and image retrieval.

Chom Kimpan received D.Eng. in Electrical and Computer Engineering from King Mongkut's Institute 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 an associate 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.