Proceedings of the 5th WSEAS Int. Conf. on Signal Processing, Robotics and Automation, Madrid, Spain, February 15-17, 2006 (pp147-152)

A Fuzzy and Rough Sets Approach for Recognition of Handwritten Thai Characters

PISIT PHOKHARATKUL[1], DARUNEE CHATCHAWALANONTH[2], and CHOM KIMPAN[3]
[1]Department of Computer Engineering, Faculty of Engineering, Mahidol University,
Nakhon Pathom 73170, Thailand, E-mail: egpph@mahidol.ac.th
[2,3]Faculty of Information Technology, Rangsit University, Patumtani, 12000, Thailand

Abstract: - This paper presents both fuzzy logic and rough logic as applied to Thai character recognition. Both fuzzy and rough sets have been introduced as tools to deal with vagueness and uncertain data in artificial intelligence applications. They both attack the problem of impreciseness, but in different ways. Whereas fuzzy logic is concerned with vagueness, rough logic concerns itself with indiscernability. Hence rough logic views impreciseness as lack of knowledge and not as a feature of the problem itself. In a recognition system, the rough set theory can only deal with discrete values. This means that the value attributes need to be discrete before they are provided to the rough set theory. Then attempt were to use the concept of fuzzy as a tool of discretization of the features of handwritten Thai characters. The features used for the method are heads of Thai character (loop contours), end points, peripheral shape features, stroke density features, width-height ratio, feature code, number of island, characteristic head, character ripple, and chain code respectively. Rough logic offers a variety of desirable features, most notably the ability of reduction of knowledge. This enables us to be able to extract the essential information from a given rule base, and thus it offers a simple way of obtaining recognition with a minimum set of rules. The system is tested with unknown samples, composed of 53,400 handwritten Thai characters. Experimental results have shown that both fuzzy logic and rough logic are powerful tools in successfully classifying handwritten Thai characters. The recognition rate by this method is about 84 %.

Key-Words: - Handwritten Thai character, Character recognition, Fuzzy sets, Rough sets

1 Introduction

In research of the topic of Thai character recognition in the past twenty two years, many approaches have been tried to make a computer recognize the characters [1-7]. The recognition rates of these methods were satisfactory, but they have many features or over information from the system.

This research proposes a method of recognition of Thai characters using fuzzy and rough sets to choose the smallest subset of these conditional features, which remains consistent with respect to the decision feature. The idea is to transform a set of features in a set of rules that represent the recognition process of a system. The motivation of fuzzy and rough sets in the proposed hybrids has two goals:

- Construction of fuzzy values for primitive elements.

- Rough set attribute reduction is used to extract the knowledge from a domain in a concise way, which retains the information content whilst reducing the amount of knowledge involved.

The recognition procedure will be illustrated in the following section.

2 Thai character recognition system

Thai characters consist of 44 consonants, 18 vowels, 4 controlled voice tones, 3 special symbols, and 10 Thai numerals, as shown in figure 1. As a result, 89 fundamental characters (including Thai numerals and Arabic numerals) are obtained. The structure of most Thai characters consists of small loops (head of character) combined with curves and lines.

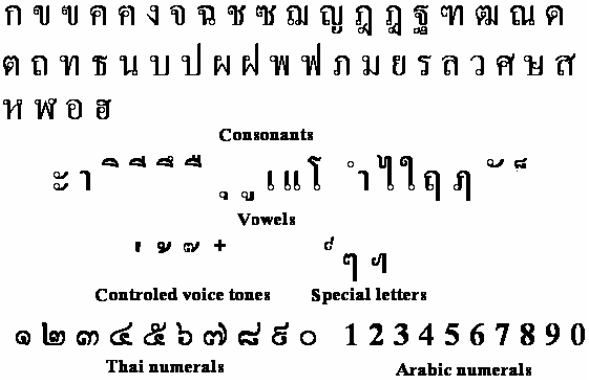

Fig.1 Thai character set consists of 44 consonants, 18 vowels, 4 controlled voice tones, 3 special letters, 10 Thai numerals, and 10 Arabic numerals.

In general, a Thai language sentence is composed of consonants, vowels, and tonal symbols on different levels. The vertical level can be divided into four parts.

A typical character recognition system consists of the following stages as shown in figure 2. The training and testing handwritten Thai characters were scanned in binary form at a resolution of 600 dpi. We used 26,700 trained characters for training (from 100 peoples, each person made 3 copies of sheets), and 53,400 unknown characters for testing.

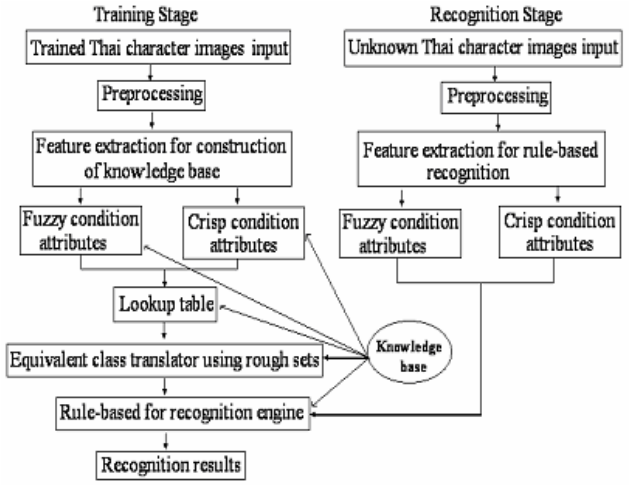

Fig. 2 Topology of rough logic recognizer

The preprocessing stage involves noise removal, smoothing borders of character, character image segmentation, and zoning references respectively.

In the feature extraction, the features of each character are extracted and saved to a file for the next step. These features of characters reveal that some of the condition attributes are vaguely defined by fuzzy sets, and some condition attributes can be discrete by crisp sets. Thus the equivalent classes may be difficult to distinguish since their boundaries are vague. This leads to the idea of augmenting the rough recognizer with a fuzzy subsystem. The features are split into fuzzy condition attributes and crisp condition attributes. The fuzzy condition attributes and crisp condition attributes are collected into the lookup table. The rough sets theory works to reduce the dataset and remains consistent with respect to the decision feature as a filter-based tool. So, the principle of the rough set is based on the assumption that with every object some information (data, knowledge) can be associated, and knowledge in the data is merged according to debasing the precision of the data. Accordingly it takes on favourable effects to detect knowledge from imprecise information. How to apply rough set theory to making recognition system is shown in the next section.

3 Features Extraction

3.1 Zones of Reference

Each character will be normalized to 128×128 pixels with the same width-height ratio, and computed a centroid of character. Then the character is divided into horizontal and vertical zones following the line drawn to pass the centroid. The names given to each zone are S_h^1, S_h^2, S_h^3, …, S_h^8 and S_v^1, S_v^2, S_v^3, …, S_v^8 respectively.

3.2 Head and end point of Thai character [5]

The heads and end points are the distinctive features of Thai characters. They appear in various locations of each character. The attributes Head_Zone_Z_n and End_point_z_n are defined by whether the zone z_n have them or not. The possible values of these attributes are True or False (have or none). The broken heads are not a problem, because it uses fuzzy sets measure the portion of loop contour.

3.3 Peripheral shape features [8, 9]

The character image input is first divided into 8 sections. In each section we finding the distances from the character to the edge of the section in both the horizontal and vertical and apply the equation (1), (2)

$$PA_h^i = (\sum_{j=1}^{k} d_h^j) / area \quad (1)$$

$$PA_v^i = (\sum_{j=1}^{k} d_v^j) / area \quad (2)$$

where PA_h^i is the horizontal peripheral background area,

PA_v^i is the vertical peripheral background area,

d_v^j is the distance between the outermost stroke edge and the character image and *area* is the subarea of the i th in the character image.

After that we calculate the horizontal peripheral line difference and vertical peripheral line difference from the results of equation (1), (2), by the equation (3), (4)

$$PD_h^i = \{\sum_{j=1}^{k} (\| d_h^j \| - \| d_h^{j+1} \|)\} / area \quad (3)$$

$$PD_v^i = \{\sum_{j=1}^{k} (\| d_v^j \| - \| d_v^{j+1} \|)\} / area \quad (4)$$

where PD_h^i is the horizontal peripheral line difference, and
PD_v^i is the vertical peripheral line difference.

The horizontal peripheral shape features of each subarea are shown in figure 3 a.).

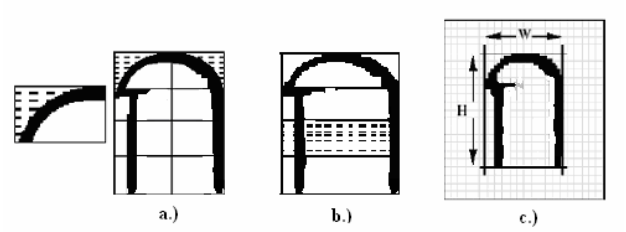

Fig. 3 a.) Peripheral shape features.
b.) Stroke density features.
c.) Width (W) and Height (H) of character image.

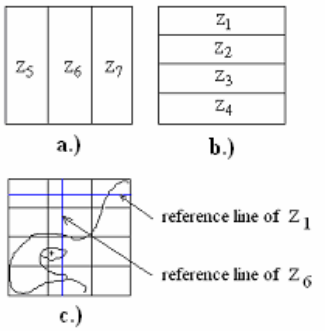

Fig. 4 (a) The zones that use vertical referent lines.
(b) The zones that use horizontal referent lines.
(c) Two referent lines pass Zones Z_1 and Z_6 that make an attribute to be Feature_Code_Z_1= 1 and Feature_Code_Z_6 = 5.

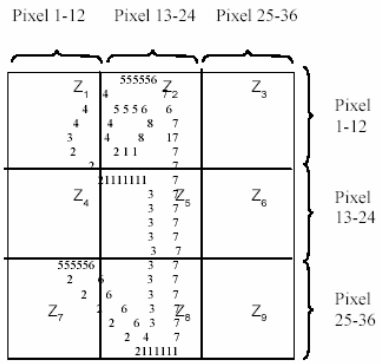

Fig. 5. Chain code of character image.

3.4 Stroke density features

The stroke density feature is another method that is good to represent the character's structure. In this work, 2 sets of features are computed by this means: one is the horizontal stroke, the other is the vertical stroke. Each set is composed of 4 features, as shown in figure 3 b.).

Those stroke features are computed by equation (5), (6)

$$SD_h^i = \sum_{i=1}^{k} \sum_{y} \overline{s(x,y)}.s(x,y+1)/m \quad (5)$$

$$SD_v^i = \sum_{i=1}^{k} \sum_{x} \overline{s(x,y)}.s(x+1,y)/m \quad (6)$$

where SD is horizontal stroke density,
SD is vertical stroke density, and
$s(x, y)$ is a pixel at (x,y) position.

3.5 Width-High Ratio

Figure 3 c.) shows width and height of the character image. We calculate the ratio between width and height of character image by equation (7)

$$\text{RateWH} = W/H \quad (7)$$

3.6 Feature code

The feature code is defined by the maximum number of points that the referent lines pass in its zone. 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.

3.7 Chain Coding

In order to compute the chain code, a character image is divided into nine equal sub regions by two vertical lines and two horizontal lines as shown in Figure 5. The chain code of a character image is computed as representing and we count the codes in each sub region. In counting, we defined the codes 1 and 5 as representing a code 1, 2 and 6 as representing a code 2, 3, and 7 as representing a code 3, and 4 and 8 as representing a code 4 respectively. All of the chain codes in each sub region are required to evaluate the main code (maximum code). The sub region has no chain code, we represent it as a code 0.

3.8 Number of Island

Number of island is a number of isolated parts of the letter. An attribute called *No_Island* is used to measure number of islands as shown in figure 6 a.), and 6b.).

3.9 Characteristic Head

In similar Thai characters it is necessary to use characteristics of heads. The characteristic head has more than one type of curl such as an inner curl and outer curl as shown in figure 6c.), 6d.).

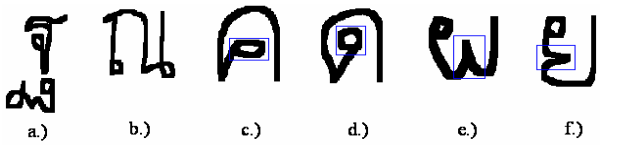

Fig. 6 a.) Number of Islands: <*No_Island* = 2>.
b.) Number of Islands: <*No_Island* = 1>.
c.) characteristic head: Outer curl.
b.) characteristic head: Inner curl.
e.) Vertical ripple.
f.) Horizontal ripple.

3.10 Character Ripples

In some Thai character, the ripple feature has more than one type of position. Figure 6 e.), and 6 f.) show an example of vertical and horizontal ripples.

4 Fuzzy modeling for Pattern Recognition

Fuzzy set theory and fuzzy modeling techniques have been applied to various fields such as pattern recognition, fuzzy control system, fuzzy expert system and data clustering. These techniques allow objects to be assigned to different regions or clusters to different degrees, thereby often improving the model of the data structure. The features of characters images are modeled by linguistic variables such as LOW, MEDIUM, HIGH, etc. and then the letter is used within a predetermined set of fuzzy rules.

A linguistic variable was characterized by a quintuple [10]

$$(x, T(x), U, G, M) \qquad (8)$$

where x is the name of variable,
U is the universe of discourse,
T(x) is a set of terms of a nature or artificial language used to speak about x,
G is the syntactic rule used to generate the terms of T(x), and
M is the semantic rule defining the meanings of T(x).

This semantics associated each term x of T(x) with the base variable U according to the compatibility $M_{RX}(U)$ of U with the fuzzy set T(x). Each fuzzy set T(x) is defined by the corresponding restriction $R_x(x)$ associated with each term x of T(x).

If the fuzzy system has n input $(x_1, x_2, \ldots, x_n)T$ and m output $(y_1, y_2, \ldots, y_n)T$, the system is composed of fuzzy rules in the form:

IF (x_1 is A_1^i) and … and (x_n is A_n^i) THEN (y is B^j)

where A_k^i , k= 1, .., n are linguistic variables which represent vague terms such as SMALL, MEDIUM or LARGE defined on the input and B^j are output variables, respectively.

A rough set deal with discrete values but the data values of some condition attributes are continuous. The clustering method was used to quantify data values of each attribute. The clustering task classifies data into more-or-less homogeneous classes. The similarity measure was important for clusters of similar data in the homogenous class. In fact, data values of each attribute are more vague and it is difficult to find the boundary of each class. From this problem, we use fuzzy clustering to provide the splitting results and combine the cluster membership degree. A fuzzy c-Means algorithm is fuzzy clustering, and provides an iterative approach to classifying data to a class, and approximating the boundary of each class for fuzzy structure. The clustering result is best identified when more points concentrate around the cluster center.

5 Rough Sets Theory

Rough sets [11, 12] have been introduced as a tool to deal with inexact, uncertain or vague knowledge in artificial intelligence applications. The idea of rough sets works with lower and upper approximation. They are interior and closure operations in a certain topology. A major area of application of rough set theory is the study of dependencies among attributes of information systems. An information system S = (U, A), where U was a nonempty, finite set of objects known as the universe, and A is a finite of attributes. For every element there exists a set of its values known as a domain of *a*, denoted as V*a*. For every attribute $a \in$ A (*a* belongs to A), there exists a function u: U $\rightarrow$ V*a*, which assigns a unique attribute value from V_a to every object x $\in$ U. These are now considered subsets of attributes B $\subseteq$ A, where B uniquely defines equivalence relations.

$$IND(B) = \{x, y) \in U^2 : a(x) = a(y) \quad for \quad every \quad a \in B \qquad (9)$$

The family of all equivalence classes of the equivalence relation IND (B) was denoted U/IND(B). With every X $\subseteq$ U, and B $\subseteq$ A we associate two sets defined as follows:

$$\underline{B}X = \cup\{Y \in U \;/\; IND(B) : Y \subseteq X\} \qquad (10)$$

was the lower approximation or positive region of X,

and

$$\overline{B}X = \cup\{Y \in U / IND(B) : Y \cap X \neq \phi\} \quad (11)$$

was the upper approximation or possible region of X.

A set $BN_B = \overline{B}X - \underline{B}X$ will be called B-boundary of X.

If $X \subseteq U$ is given by a predicate P and $x \in U$, then

1. $x \in \underline{B}X$ means that x certainly has property P,
2. $x \in \overline{B}X$ means that x possibly has property P,
3. $x \in U \backslash \overline{B}X$ means that x definitely does not have property P.

The area of uncertainly extends over $\overline{B}X \backslash \underline{B}X$, and the area of certainty is $\underline{B}X \cup \text{-}\overline{B}X$, respectively.

5.1 Core and Reduct

If such a decision attribute D depends on a subset of condition attributes C and minimum C is C′, while D depends on C′ too. We call C′ a reduct of C. The intersection of all reducts of C is called the core of C. Unless C has only of reduct, the core of C is not itself a reduct.

5.2 Core computation from discernibility matrix

A discernibility matrix of C in S, M(C) = {mij}n×n as shown in equation (12)

$$(m_{ij}) = \begin{cases} \phi & x_i, x_j \in \text{same concepts} \\ \{c \in C : f(c,x_i) \neq f(c,x_j)\} & x_i, x_j \in \text{different concepts} \end{cases} \quad (12)$$

m_{ij} contains the attributes whose value are not identical on both xi and x_j (x_i, x_j belong to different classes, that is x_i, x_j represent different concepts).

5.3 Compute the best reduct or a minimal attribute subset

The best reduct of a minimal attribute subset is based on the dependency relationship and the significant values of attributes. So, the best reduct contains minimal attributes not dependent on each other. The degree of dependency k is defined as

$$k = \frac{|POS_c(D)}{|U|} \quad (13)$$

where D and C is a subset of total attributes A, U is a set of universal relationship, and

$$POS_c(D) = \bigcup_{x \in U / I(D)} C_*(X) \quad (14)$$

called a positive region of the partition U/D with respect to C, is the set of all elements of U.

5.4 Decision rules

The numbers of correct rules in a decision table are measured by r(C, D), where C and D are the condition attributes and decision attributes respectively. Define

$$\gamma(C,D) = \frac{POS_c(D)}{|U|} \quad (15)$$

γ(C, D) is a relative frequency of the number of correct rules. If γ(C, D) = 1, the precision success is perfect. The decision rules are often presented as implication, and are often called "If…Then…" rules. If a, b, c, d are the condition attributes of set A, D are the decision attributes, we could express the rules as follows:

Rule I
  IF a = L THEN D = 1 (for the class I)
Rule II
  IF b = S AND c = M THEN D = 2 (for the class II)
  ……. …….
Rule n
  IF d = L THEN D = n (for the class n)

We set up an object's attributes from multi-features in each class. The intervals of attributes discretized to fuzzy sets S, M and L, etc., defined by fuzzy C-means algorithm. Then we apply the rough sets to the decision table, and compute the best reduct or user minimal attribute subset using the reduct algoritm in [12]. The finding of the best reduct is based on the dependency relation and the significant values of attributes as mentioned in section 5.3. From the reduct set, we can express the rough rule-based decision, and use this to classify the handwritten Thai characters from others in each group.

6 Experiment Results

In this system, there are two type condition attributes: fuzzy condition attributes and crisp condition attributes. The recognition process can thus be described as a decision table where fuzzy condition attributes and crisp condition attributes that make out the set of condition attributes and equivalence class (C) are the set of decision attributes. The fuzzy condition attributes are peripheral shape features, stroke density features, and width-height ratio. The crisp condition attributes are heads of character, end points, feature codes,

chain codes, number of island, characteristic head, and character ripples. From the decision table, we are now ready to start reducing superfluous columns, something which is done to reduct of condition attributes. To this end we employ the method described in section 5. When all superfluous attributes have been removed, we reduce the decision table further by removing superfluous values of condition attributes from the table. This means that we are searching for only those attribute values which are really necessary to distinguish all the decision classes.

Using the technique described in section 5.4, we are able to create the rough rule-base for a recognition system.

In the particular experiments the training sets were composed of 26,700 characters. A fuzzy C-means algorithm was used to quantify the attributes and rough sets, are used to create a rough decision rule-based for classification of the characters. The training and testing were scanned in binary form at a resolution of 600 dpi. The testing process was applied to 53,400 characters. From the results, the average recognition rate has about 84% accuracy. The performance of this method decreased slightly for smaller characters. Table 1 compares the average recognition rate of the fuzzy structural and rough sets approach with the structure tree approach.

Table 1 Average recognition rate of proposed system and structure tree system.

Recognition engines	Average recognition rate (%)
Fuzzy and rough sets	84
Ant-Miner [7]	81
Structure tree [5]	70

7 Conclusion

In this specific experiment, a fuzzy C-means is used to discretize attributes to construction of fuzzy structural and rough sets used to classify a character from an other using rough decision rule base. The rough sets and fuzzy sets capture different aspects of imperfect knowledge: indiscernability and vagueness respectively. The rough sets use concept of core and reduct of knowledge as a filter-based tool. The fuzzy structural and rough sets approach in this paper was better than the structure tree in reference [5], because it used the multi-features to choose the necessary attributes for the recognition system. Whereas the structure tree uses the crisp condition attributes only, and do not have the concepts of vagueness and impreciseness. If it has a defect features may lead to unrecognized problems. Both fuzzy sets and rough sets could achieve 84 % recognition performance with ideal scanning input in comparison to 70 % of the structure tree used.

References:

[1] C. Kimpan, A. Itoh, and K. Kawanishi, "Recognition of printed Thai character using a matching method", *Proc. IEE,* vol.130, Pt.E, No.6, 1983, pp. 183-188.

[2] C. Kimpan, "Printed Thai character recognition using topological properties method.", *INT.J. Electronics*, vol.60, No.3, 1986, pp. 303-329.

[3] C. Kimpan, A. Itoh, and K. Kawanishi, "Fine classification of printed Thai character recognition using the Karhunen-Loeve expansion", *Proc. IEE*, vol.134, Pt.E., No5, 1987, pp. 257-264.

[4] P. Hirananichakorn, T. Agui, and M. Nakajima, "A recognition method of handprinted Thai characters by local features.", *IECE trans.,* vol. E68, No.2, 1985, pp. 83-90, Feb. 1985.

[5] S. Airphaiboon and S. Kondo, "Recognition of handprinted Thai Characters Using Loop structures.", *IEICE Trans. INF.&SYST.*, Vol.E79-D, No.9, 1996, pp. 1296-1304.

[6] P. Phokharatkul, and C. Kimpan, "Handwritten Thai Character Recognition Using Fourier Descriptors and Genetic Neural Networks.", *Computational Intelligence An International Journal*, Vol. 18, No. 3, 2002, pp.270-293.

[7] P. Phokharatkul, K. Sankhuangaw, S. Somkuarnpanit, S. Phaiboon, and C. Kimpan, "Off-Line Hand Written Thai Character Recognition using Ant-Miner Algorithm.", *Enfomatika*, Vol. 8, 2005, pp. 276-281.

[8] T. Suebsanit, and C. Kimpan, "Printed Thai Character Recognition using Multifeature and Multilevel Classification", *SCORED 2001*, 2001, pp. 167-170.

[9] Yuan Y. Tang, and Lo-Ting Tu. "Off-line recognition of Chinese handwriting by multifeature and multilevel classification", *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 20. No. 5, 1998, pp. 556-561.

[10] G.J. Klir, U.St. Clair, and B. Yuan, *Fuzzy Set Theory Foundations and Applications*, Printice Hall international, Inc., 1997.

[11] Z. PAWLAK, "Why rough sets?", *Proceedings of the Fifth IEEE International Conference on Fuzzy Systems*, 1996., Volume: 2 , 1996, pp. 738 -743.

[12] X. Hu**.** ***"Knowledge Discovery in Database: An Attribute-Oriented Rough Set Approach"*** Ph.D. Thesis of University of Regina. 1995.