Note that it is important that the exact values of x^k be used rather than the quantized ones.

References

[1] T. Pavlidis, "Linguistic analysis of waveforms," in *Software Engineering*, vol. 2, J. T. Tou, Ed. New York: Academic, 1971, pp. 203–225.
[2] D. E. McClure, "Feature selection for the analysis of line patterns," Brown Univ., Providence, R.I., Tech. Rep., May 1970.
[3] T. Pavlidis, "Waveform segmentation through functional approximation," *IEEE Trans. Comput.*, vol. C-22, pp. 689–697, July 1973.
[4] T. Pavlidis and S. L. Horowitz, "Piecewise approximation of plane curves," in *Proc. Int. Joint Conf. Pattern Recognition*, Washington, D.C., Oct. 1973.
[5] S. L. Horowitz, "A general peak detection algorithm with applications in the computer analysis of electrocardiograms," in *Proc. Second Int. Joint Conf. Pattern Recognition*, Copenhagen, Denmark, Aug. 1974.
[6] D. E. McClure, "Nonlinear segmented function approximation and analysis of line patterns," Brown Univ., Providence, R.I., Tech. Rep., 1973.
[7] C. L. Lawson, "Characteristic properties of the segmented rational minimax approximation problem," *Numer. Math.*, vol. 6, pp. 293–301, 1964.
[8] T. Pavlidis and A. P. Maika, "Uniform piecewise polynomial approximation with variable joints," *J. Approx. Theory*, vol. 12, 1974, (in press).
[9] T. Pavlidis, "Optimal piecewise polynomial L_2 approximation of functions of one and two variables," Princeton Univ., Princeton, N.J., Tech. Rep. 143, Oct. 1973.
[10] J. R. Blum, "Multidimensional stochastic approximation methods," *Ann. Math. Statist.*, vol. 25, pp. 737–744, 1954.
[11] N. V. Loginov, "Methods of stochastic approximation," *Automat. Remote Contr. (USSR)*, vol. 27, pp. 706–728, 1966.
[12] K. S. Fu, *Sequential Methods in Pattern Recognition and Machine Learning*. New York: Academic, 1968.
[13] A. Katopis, "Multidimensional stochastic approximation and its application to detection and estimation," Ph.D. dissertation, Dep. Elec. Eng., Princeton Univ., Princeton, N.J., Dec. 1972.
[14] T. Pavlidis, "On the convergence of a scheme for uniform polynomial approximation with variable joints," Princeton Univ., Princeton, N.J., Tech. Rep. 140, Sept. 1973.

Fuzzy Logic for Handwritten Numeral Character Recognition

PEPE SIY, MEMBER, IEEE, AND C. S. CHEN, MEMBER, IEEE

***Abstract*—A recognition system based on the concept of fuzzy logic for handwritten numeral character recognition is presented. A handwritten character is considered as a directed abstract graph, of which the node set consists of tips, corners, and junctions, and the branch set consists of line segments connecting pairs of adjacent nodes. The line segments (branches) are fuzzily classified to branch types (features) such as straight lines, circles or portions of circles. Since the features under consideration here are fuzzy in nature, the fuzzy set concept is utilized, and the features are treated as fuzzy variables. A fuzzy variable is characterized by a membership grade function the value of which ranges in the closed unit interval. The two extreme membership grades of zero and unity indicate nonmembership and full membership status of the branch under consideration. Handwritten characters are considered ill-defined objects. With the aid of fuzzy functions such objects can be objectively defined and studied. A handwritten character is represented by the fuzzy function, which relates its fuzzy variables, and by the node pair involved in each fuzzy variable. The process of recognition is accomplished in two steps. First, the unknown character is preprocessed to produce its representation and second, the classification of the unknown character is reduced to finding a character (previously learned) of which the representation is isomorphic to the representation of the unknown character. Experimental results using IEEE Pattern Recognition Data Base No. 1.2.1 Handprinted Numeric Characters are presented. A recognition accuracy of 98.4 percent has been attained.**

Manuscript received January 16, 1974; revised May 29, 1974.
P. Siy was with the Department of Electrical Engineering, University of Akron, Akron, Ohio 44235. He is now with the Burroughs Corp., Detroit, Mich.
C. S. Chen is with the Department of Electrical Engineering, University of Akron, Akron, Ohio 44325.

I. Introduction

Pattern recognition is an area that requires interdisciplinary collaboration among engineers, physicists, physiologists, psychologists, logicians, mathematicians, and philosophers. Its objective is the mechanization of human perception and concept formation. It is known that the brain does not store a picture of the world, but some aspects of it [1]. Recognition of objects is then based on these aspects or features and not on hard copy. Neurologists have been trying to crack the neural codes to shed light on the inner workings of the brain. Hubel and Wiesel of Harvard University, Cambridge, Mass., have disclosed some of the features that are coded in the neurons, such as lines and edges of various orientations, junctions, end points, etc. [2], [10].

In this correspondence a handwritten character recognition system is proposed that incorporates: 1) the neurological findings of Hubel and Wiesel [2] as the natural feature set, 2) the psychological findings of Pavlov [1] on stimulus generalization as the means of features reduction, 3) the fuzzy set concept introduced by Zadeh [3] in 1965 as the tool to resolve the great distortion found in handwritten characters, and 4) the structural approach of pattern recognition as the tool to achieve size and position invariant properties.

Features that include elements of a node set and elements of a branch set, which are similar to those used by several researchers [11], [13] are selected. The lack of precision in the definitions of the elements of the feature set necessitates the use of the fuzzy set concept. The abstract graph concept is then used to characterize the network of fuzzy sets, which represent the handwritten character under consideration. This characterization is position, size, and distortion invariant.

The structure and operation of the character recognition system is presented in the next section. Preprocessing, membership assignment of each feature element, and the pattern classifier are discussed in detail. Section IV presents the results of testing the system using the IEEE Pattern Recognition Data Base No. 1.2.1. This standard data base was selected in order that different recognition systems may be compared.

II. Handwritten Character Recognition System

The system consists of two basic units: A) features extraction and the handwritten character representation, and B) the pattern classifier.

A. Handwritten Character Representation

Handwritten characters are a distorted variant of printed characters; therefore, any study of handwritten characters must start with printed characters. There are essentially three basic elements of alpha-numeric characters: 1) the straight line (vertical, horizontal, and slant), 2) the circle, and 3) a portion of a circle of various orientations [4].

In 1965, L. A. Zadeh [3] introduced the "fuzzy set" concept. A fuzzy set is a class of objects with a continuum of grades of membership. Such a set is characterized by a membership (characteristic) function, which assigns to each object a grade of membership ranging between zero and one. The class of hand-drawn lines, circles, or portions of circles are considered as fuzzy sets in this correspondence.

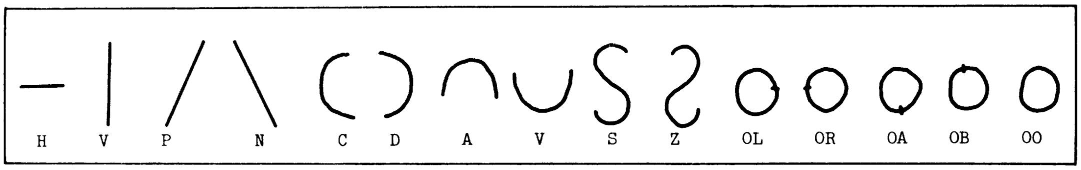

Fig. 1. The branch feature set.

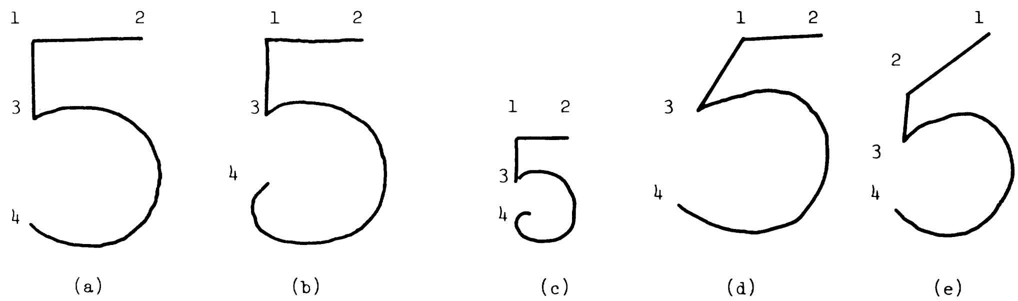

Fig. 2. Typical examples of character 5.

The following features are found sufficient for handwritten numerals, many of which have also been used by other authors [11].

1) Straight line features:

horizontal	(H)
vertical	(V)
positive slope	(P)
negative slope	(N)

2) Portion of a circle features:

C curve	(C)
D curve	(D)
A curve	(A)
V curve	(V)
S curve	(S)
Z curve	(Z)

3) Circles:

circle on the left of its node	(OL)
circle on the right of its node	(OR)
circle above its node	(OA)
circle below its node	(OB)
circle	(OO)

Let $B = \{H,V,P,N,C,D,A,V,S,Z,OL,OR,OA,OB,OO\}$ denote the branch feature set shown in Fig. 1.

The patterns in Fig. 2(a)–(c) are not exactly alike, however, their functional descriptions are the same, i.e., $F(5a) = F(5b) = F(5c) = H(1,2) \cdot V(1,3) \cdot D(3,4)$. Two patterns are said to be equivalent if their functional descriptions are the same. Thus $F(5) = H(1,2) \cdot V(1,3) \cdot D(3,4)$ defines an equivalent class in which the characters in Fig. 2(a)–(c) are typical members. This is the same as the idea of generalization, which means that one has only to learn one typical example of character 5 in order to recognize any equivalent variants. This definition of pattern (or stimulus) equivalence is both size and position invariant, which is compatible with visual perception in man. Other variants of the character 5 of which the functional descriptions are different are shown in Fig. 2(d) and (e). They are $F(5d) = H(1,2) \cdot P(1,3) \cdot D(3,4)$; $F(5e) = P(1,2) \cdot V(1,3) \cdot D(3,4)$. Using the fuzzy logical OR operation, one can combine the various descriptions of a character. For example, the character 5 in Fig. 2 can be represented by

$$\begin{aligned} F(5) &= F(5a) + F(5b) + F(5c) + F(5d) + F(5e) \\ &= H(5,2) \cdot V(1,3) \cdot D(3,4) \\ &\quad + H(1,2) \cdot P(1,3) \cdot D(3,4) \\ &\quad + P(1,2) \cdot V(1,3) \cdot D(3,4). \end{aligned}$$

B. Pattern Classifier

The decision criteria used for pattern classification are executed in two steps.

1) The branch features of the pattern to be classified and the branch features of the prototypes of each class are compared. Those prototypes that perfectly match the branch features of the pattern are retrieved.

2) The node pairs of the same branch type of the pattern to be classified and each retrieved prototype are compared. The fact that the numbering of the nodes in the pattern and prototypes may not be the same implies that an isomorphic mapping has to be found. This can be done as follows. Let $(n_\phi'(bi),n_\theta'(bi))$ and $(n_\phi(bi),n_\theta(bi))$ denote the node pair of branch type bi, $i = 1,\cdots,n$, of the pattern and prototype under consideration, respectively. The mapping

$$\Psi = \{(n_\phi'(bi),n_\phi(bi)),(n_\theta'(bi),n_\theta(bi) \mid k = 1,\cdots,n\}$$

is defined. When this is isomorphic, the prototype is accepted; when this is not, it is rejected.

III. The Recognition System

The block diagrams of the recognition system for the learning phase and the recognition phase are shown in Fig. 3(a) and 3(b), respectively. The main features of these diagrams consist of the following components: input pattern, thinning, labeling, coding, prototype memory, and pattern classifier. Each component is described in the following.

A. Input Pattern

The input pattern of the system must be digitized into a rectangular picture-frame array $P = \{p = (i,j) \mid 1 \leq i \leq n, 1 \leq j \leq m;\ n,m \in N\}$. The pattern is a binary picture; that is,

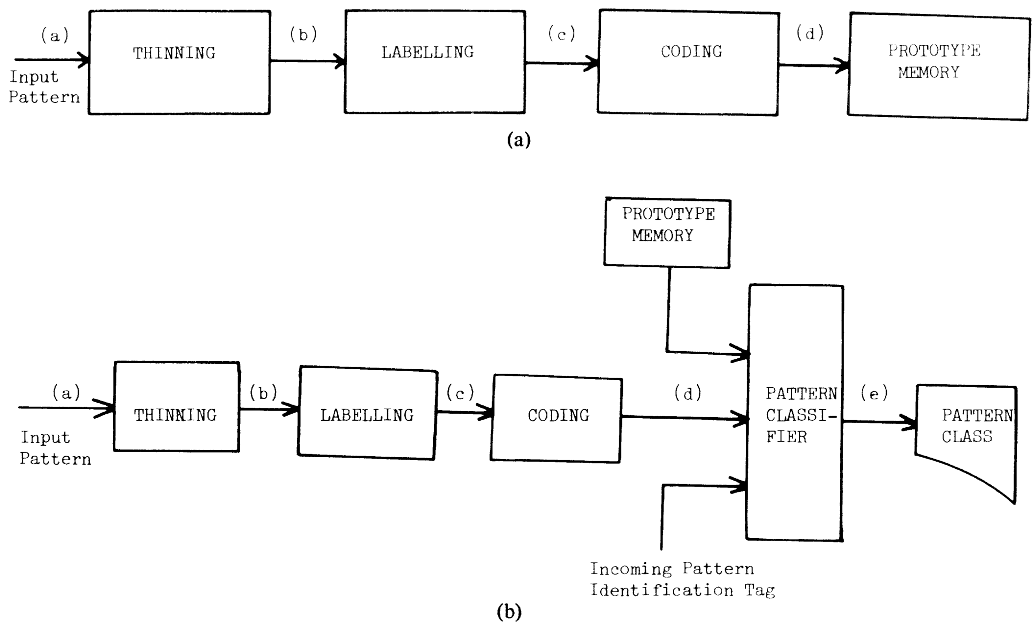

Fig. 3. Block diagram of recognition system. (a) Learning phase. (b) Recognition phase.

the points on the pattern assume the value of one while the other points of the frame take the value of zero.

B. Thinning

The skeleton of a pattern is obtained by using Hilditch's thinning algorithm [6]. In this work the algorithm is restricted for binary patterns. Basically, the algorithm tests the boundary points of the pattern. Those points on the boundary the removal of which would not alter connectivity and those that do not lie on a tip of a line are deleted.

C. Labeling

1) Node Detection and Labeling: A node set in the skeleton of a pattern is defined as the collection of tips (points that have one neighbor), corners (points that have two neighbors and where an abrupt change of line direction occurs), and junctions (points that have three or more neighbors).

2) Branch Detection and Labeling: A branch is a line segment connecting a pair of adjacent nodes. For the degenerate case of a line pattern without nodes, the pattern is considered as a circular branch. A branch of which the length is less than threshold S_B is considered extraneous and consequently removed by declaring its nodes equivalent.

In classifying branches, two sources of fuzziness are attributed to the measure of a) straightness and b) orientation. Both sources are present in noncircular branches and only the second source is present in circular branches with node.

a) Measure of straightness: The measure of straightness of a branch is determined by fitting a straight line with the minimum least squares error, and the branch is represented by the line with the least squares error.

A measure of straightness for a noncircular branch can be defined by

$$f_{SL} = 1 - S/S_T, \qquad \text{if } S < S_T$$
$$= 0, \qquad \text{if } S \leq S_T$$

where S_T is the threshold least squares error. A given branch is classified as a portion of a circle, if $0 \leq f_{SL} < 0.5$; a straight line, if $0.5 < f_{SL} \leq 1$; either, if $f_{SL} = 0.5$.

b) Measure of orientation: If the classification of a branch to the class of a straight line, a portion of a circle, or a circle with node is known, then the measure of orientation can be used to further characterize this branch.

Straight line class: This class contains the sets H,V,P,N. To classify a given branch into one of these sets, a measure of orientation based on the slope $\theta = \tan^{-1} m$ of the best fit line is required. With this slope the following membership grade functions for the sets H,V,P,N are defined as

$$FH(\theta) = 1 - \min\{\min[|\theta|, |180 - \theta|, |360 - \theta|]/45, 1\}$$

$$FV(\theta) = 1 - \min\{\min[|90 - \theta|, |270 - \theta|]/45, 1\}$$

$$FP(\theta) = 1 - \min\{\min[|45 - \theta|, |225 - \theta|]/45, 1\}$$

$$FN(\theta) = 1 - \min\{\min[|135 - \theta|, |315 - \theta|]/45, 1\}.$$

The branch is classified to the set with the highest membership grade assignment.

Portion of circle class: This class contains the sets C,D,A,V,S,Z. These sets can be further grouped as vertical (C,D), horizontal (A,V), or either (S,Z). To assign a given branch into one of the above groups in accordance with its measure of orientation, the slope $\theta = \tan^{-1}(y_n - y_1)/(x_n - x_1)$ of the line connecting the end points of the branch is considered. To accomplish this, two fuzzy sets are considered, namely, the horizontal curve (HC) and the vertical curve (VC). Based on the slope, the membership grade functions for HC, VC are defined as

$$FHC(\theta) = 1 - \min\{\min[|\theta|, |180 - \theta|, |360 - \theta|]/90, 1\}$$

$$FVC(\theta) = 1 - \min\{\min[|90 - \theta|, |270 - \theta|]/90, 1\}.$$

Consequently, the branch is classified to the set with the highest membership grade assignment. The group classified as either (S,Z) depending on its orientation is treated as HC or VC.

The branch is further characterized to one of the elements of each group by considering nonfuzzy information. This is accomplished by drawing two lines parallel to the LN line connecting the end points. The construction to be discussed is illustrated in Fig. 4. For a horizontal curve, one of the lines is drawn above the LN line and the other below, and they are

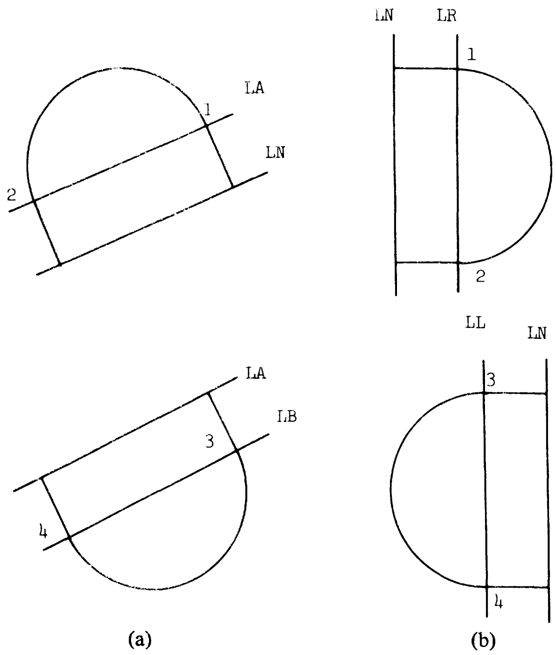

Fig. 4. Line construction for classifying given branch into class of portion of circle. (a) For horizontal curves. (b) For vertical curves.

denoted as LA and LB, respectively. For a vertical curve, the lines are drawn to the left and right of the LN line and are denoted as LL and LR, respectively. The number of points of intersection between the lines LA,LB,LR,LL, and the branch are denoted by IA,IB,IR,IL, respectively. The points of intersection of the branch with LA (or LR) are labeled as $(x_1,y_1),(x_2,y_2)$, and with LB (or LL) as $(x_3,y_3),(x_4,y_4)$. The conditions for a branch (b) to belong to one of the sets C,D,A,V,S,Z are

$$C = \{b \mid b \in VC \wedge IR = 0 \wedge IL = 2\}$$

$$D = \{b \mid b \in VC \wedge IA = 2 \wedge IB = 0\}$$

$$A = \{b \mid b \in HC \wedge IA = 2 \wedge IB = 0\}$$

$$V = \{b \mid b \in HC \wedge IA = 0 \wedge IB = 2\}$$

$$S = \{b \mid [b \in VC \wedge IR = 2 \wedge IL = 2 \wedge \min(y_1,y_2) < \max(y_3,y_4)] \vee [b \in HC \wedge IA = 2 \wedge IB = 2 \wedge \max(x_1,x_2) > \min(x_3,x_4)]\}$$

$$Z = \{b \mid [b \in VC \wedge IR = 2 \wedge IL = 2 \wedge \max(y_1,y_2) > \min(y_3,y_4)] \vee [b \in HC \wedge IA = 2 \wedge IB = 2 \min(x_1,x_2) < \max(x_3,x_4)]\}.$$

Circle with node class: This class contains the sets OL,OR,OA,OB. To classify a given branch into one of these sets, a measure of orientation based on the slope $\theta = \tan^{-1} (y_c - y_0)/(x_c - x_0)$ of the line connecting the node (x_0,y_0) of the branch and the centroid (x_c,y_c) of the inscribed area is considered. With this slope the following membership grade functions for the sets OL,OR,OA,OB are defined as

$$FOL = 1 - \min[|180 - \theta|/90, 1]$$

$$FOR = 1 - \min\{\min[\theta, |360 - \theta|]/90, 1\}$$

$$FOA = 1 - \min[|90 - \theta|/90, 1]$$

$$FOB = 1 - \min[|270 - \theta|/90, 1].$$

The branch is classified to the set with the highest membership grade assignment.

D. Computer Coding of Characters

The functional character representation as discussed in Section IIA consists of both alphabetic and numeric codes. For example, a functional representation of character 5 is $F(5) = H(1,2) \cdot V(1,3) \cdot D(3,4)$. In order to reduce labor, the elements $H,V,\cdots,OO$ of the branch feature set described in Fig. 1 are coded by $01,02,\cdots,15$, respectively. The computer coding then consists of three strings of numbers. The first string consists of the numeric codes of the branch features arranged in ascending order. The second string consists of the node pairs arranged in accordance with the branch features of the first string, and the third string is a one-digit identification tag.

E. Prototype Memory

The computer coding of a character generates an integer string that is too long to be stored in one computer memory location. To circumvent this difficulty a matrix array with 21 columns, and a variable number of rows is allocated. Each row represents a prototype. The first string code of the prototype occupies the first ten columns, the second string code the next ten columns, and the third string code (character represented) the last column. The computer coding of character 5 is stored in the memory as 010206000012133400005.

F. Pattern Classifier

The decision criteria, as discussed in Section IIB, are simplified by computer coding into the following form. Step 1 is now changed to an equality test between the first string code of the pattern and the prototypes, while in step 2 all the prototypes retrieved from step 1 and all the second string codes of the pattern that are generated by permuting the node pairs of the nondistinct branches are tested by setting up arrays of the following form:

$$\begin{pmatrix} \text{second string code of the pattern} \\ \text{second string code of the prototype} \end{pmatrix}.$$

If at least one of the arrays tested is a permutation, the prototype is accepted; otherwise it is rejected.

IV. The System Operation and the Experimental Results

The operation of the recognition system consists of two phases: A) the learning phase and B) the recognition phase.

A. Learning Phase

The prototypes of each pattern class are learned and stored in the prototype memory. Each pattern is thinned, labeled, and coded. The code of the first pattern is stored in the prototype memory and starts a running code count of one, which means that there is one pattern classified to this code. If the succeeding code of a pattern is of the same form as the previous one, the running code count is incremented by one, while if the code is different it is stored in the prototype memory and starts a new code count. This process is repeated for the codes of all the incoming patterns, and they are accordingly allocated. Upon the end of the learning process, the memory contains all the prototype codes and their corresponding code counts. The code count represents the frequency of occurrence of each prototype code and, therefore, its relative importance compared to the other prototype codes. This can be utilized to speed up the recognition

TABLE I
LEARNING STATISTICS

Prototype Code	Identification Tag	Code Count
15000000000000000000	0	44
12000000001100000000	0	1
06000000001200000000	0	2
13000000001100000000	0	1
05000000001200000000	0	1
08000000001200000000	0	1
02000000001200000000	1	49
02020000123400000000	1	1
01060000231200000000	2	12
01010600003224120000	2	1
01010600002134130000	2	1
10000000001200000000	2	34
07100000002113000000	2	1
01061400002312220000	2	1
01060600002313340000	3	11
01070800002313430000	3	8
01060800002313430000	3	2
03060600002312240000	3	1
01060700003224120000	3	7
06070000002312000000	3	5
06000000001200000000	3	1
06060000001223000000	3	13
06080000001232000000	3	1
01060700004323120000	3	1
02020500001334230000	4	30
02020800001334230000	4	2
02050000002313000000	4	4
01020205003413352300	4	11
01020500003423130000	4	2
02030500003423130000	4	1
09000000001200000000	5	43
01020600001234450000	5	1
01090000002313000000	5	1
06000000001200000000	5	1
01060000001234000000	5	2
06070000002321000000	5	1
01061400002123220000	5	1
02140000001222000000	6	5
03140000001222000000	6	16
02020600001223230000	6	1
05120000001222000000	6	12
07120000002122000000	6	2
03120000001222000000	6	11
02120000001222000000	6	2
03141400001222220000	6	1
06000000001200000000	7	45
01030000002113000000	7	4
03070000003412000000	7	1
02030506003534132300	8	1
03050606003413352300	8	1
05050600001445460000	8	1
05061300001213110000	8	1
02050606003413352300	8	1
13140000001111000000	8	41
02051300001312110000	8	1
12140000001111000000	8	1
05061400001212220000	8	1
02061400002313330000	8	1
02130000001211000000	9	26
02020500002334130000	9	2
02110000001211000000	9	13
02030500002312120000	9	1
02030500003423130000	9	2
03130000001211000000	9	2
13000000001100000000	9	1
03030500002334130000	9	1
03110000001211000000	9	2

time by arranging the prototype codes in descending order according to their code counts.

B. Recognition Phase

The learned prototype codes are used to classify the unknown incoming patterns to the class of the matching prototype. The pattern code in the recognition phase in an actual system does not include an identification tag. However, for the purpose of generating misclassification statistics, the identification tag is required. Each incoming pattern is also thinned, labeled, and coded. Then the pattern classifier searches for the prototypes that match the pattern code. Three results can be obtained: classification, misclassification, and unclassification. Classification occurs when only one prototype is accepted and its identification tag matches that of the incoming pattern. Unclassification occurs when no prototype is accepted. Misclassification occurs when only one prototype is accepted and its identification tag is different from that of the incoming pattern.

TABLE II
SUMMARY OF CONFUSING CODES

Confusing Code	Possible Character	No. of Character Classified
06000000001200000000	0	2
	3	1
	5	1
	7	45
02020500001334230000	4	30
02020500002334130000	9	2
02030500003423130000	4	1
	9	2
13000000001200000000	0	1
	9	1

C. Experimental Results

The ideal situation occurs when the codes that characterize different pattern classes are distinct. However, in handwritten characters the ideal situation usually does not occur since different characters, because of the way a person writes, may look the same. This situation creates ambiguous characters, and a confusing code is produced. By confusing code is meant that a pattern code can be assigned to different character classes. These confusing codes must be resolved before recognition can proceed.

The data of 50 samples for each character class (totaling 500 samples) is fed into the system for learning. The results are presented in Table I. After a careful study it is seen that there are four confusing codes; these are shown in Table II. Column one of Table II indicates the confusing codes, column two represents the characters allocated to each confusing code, and column three indicates the number of times that a character is assigned to a confusing code. The confusion is resloved by attaching the dominant character to the code.

In order to increase the recognition efficiency of the system, the prototypes are arranged in descending order according to their code counts. That is, the most frequently occurring prototypes appear first on the list. These prototypes are stored in the prototype memory of the system. The input to the system is the Honeywell 500 samples with their Honeywell classification disregarded. Three samples in nine-class, two in zero-class, and one in three-, four-, and five-classes were misclassified. The rest were all classified correctly. The percentage of correct classification of this experiment is 98.4 percent, and the processing time on IBM 370 computer is 5 min 14.32 s.

V. CONCLUSION

Fuzzy logic is applied to the feature extraction of handwritten numerical characters. Dominant prototypes that are functional representations of branch features and node features are learned and stored for each character. For pattern classification, the functional representation of the incoming pattern is compared with those of the prototypes. The recognition system proposed was successfully simulated on a digital computer using the IEEE pattern recognition data base.

References

[1] K. Oatley, *Brain Mechanisms and Mind.* New York: Dutton, 1972.
[2] D. H. Hubel and T. N. Wiesel, "Receptive fields of single neurons in the cat's striate cortex," *J. Physiology*, vol. 148, pp. 574–591, 1959.
[3] L. A. Zadeh, "Fuzzy sets," *Inform. Contr.*, vol. 8, pp. 338–353, 1965.
[4] M. Dallmann, *Teaching the Language Arts in the Elementary School.* Dubuque, Ia.: Wm. C. Brown, 1966, p. 131.
[5] J. C. Hilditch, "An application of graph theory in pattern recognition," in *Machine Intelligence 3*, D. Michie, Ed. New York: American Elsevier, 1968, pp. 325–347.
[6] J. C. Hilditch, "Linear skeletons from square cupboards," in *Machine Intelligence 4*, B. Meltzer and D. Michie, Eds. New York: American Elsevier, 1969, pp. 403–420.
[7] M. A. Aiserman *et al.*, *Logic, Automata, and Algorithms.* New York: Academic, 1971, pp. 79–83.
[8] F. D. Parker, "Boolean matrices and logic," *Mathematics Magazine*, pp. 33–38, Jan.–Feb. 1964.
[9] Pepe Siy, "Fuzzy logic for handwritten character recognition," Ph.D. dissertation, Dep. Elec. Eng., Univ. Akron, Akron, Ohio, June 1973.
[10] H. B. Barlow, R. Narasimhan, and A. Rosenfeld, "Visual pattern analysis in machines and animals," *Science*, 18, vol. 177, Aug. 1972.
[11] J. T. Tou and R. C. Gonzalez, "Recognition of handwritten characters by topological feature extraction and multilevel categorization," *IEEE Trans. Comput.*, vol. C-21, pp. 776–785, July 1972.
[12] R. Narasimhan, "On the description, generation, and recognition of classes of pictures," in *Automatic Interpretation and Classification of Images*, A. Grasselli, Ed. New York: Academic, 1969, pp. 1–42.
[13] R. L. Grimsdale, F. H. Sumner, C. J. Turnis, and T. Kilburn, "A system for the automatic recognition of patterns," in *Proc. IEE*, vol. 106, Pt. B, no. 26, pp. 210–221, Mar. 1959.

Chinese Character Recognition by a Stochastic Sectionalgram Method

YUNG-LUNG MA

Abstract—**A new approach to the recognition of a block picture, for example, a Chinese character, is by comparing it with stochastic sectionalgrams obtained by grouping many samples. In order to calculate the risk, the absolute values of the differences between the stroke-occurrence probabilities of corresponding quanta in the two sectionalgrams are summed, one of these two sectionalgrams being derived from the input pattern and the other from the prototype pattern. The smaller the sum of these differences is, the more accurate the input pattern recognition. The mathematical algorithm used is new, and the method of calculation is different from ordinary ones. Moreover, the practical application to Chinese character recognition by means of dynamic programming with a computer is discussed here for the first time, although an outline was given in [1].**

I. Introduction

In [2] Fu discusses recognition of the letters of the English alphabet by using Markov chains and the algorithm of minimum risk. However, it is difficult to recognize Chinese characters by simply using these two methods. In this paper, a special sectional method is used for recognition of block pictures. The block picture, containing the whole Chinese character, is divided into arbitrary units, i.e., 20 rows and 20 columns. If we divide it into a larger number of units, we can obtain higher correct recognition rates although with longer calculation time. A layer is one row of a block picture. Four consecutive units in one layer constitute a quantum.

The principle of minimum expected risk will be used in the recognition algorithm. This risk is determined by the absolute value of the difference of the probabilities of two block pictures in corresponding quanta. If the difference of the probabilities of these two block pictures is zero, i.e., the two block pictures are the same, the risk of the measurement is also equal to zero. If the two block pictures are different, i.e., the difference of the two probabilities is nonzero, the risk is positive. This kind of risk, called the quantum risk, is to be calculated as the average risk $R(k_{i1},k_{i2},\cdots,k_{ir}; d_m)$ given by (5) and is derived as the sum of the absolute values of the differences of the probabilities $P(k_{i1},k_{i2},\cdots,k_{ir}/w_1)$ as shown in (6), where k_{ij} is the jth quantum in the ith layer.

This risk is a scalar. In order to recognize a block picture, we must add up all the risks in this block picture or in this layer as shown in the Appendix. The specified acceptable risk is defined in (A1),(A2). This specified risk can be changed. If the value of the specified risk is small, the accuracy of recognition is increased. If this value is equal to zero, for example, the accuracy of recognition is very high.

The formation of the sequential process is determined by the probability of strokes appearing in the units in each quantum of each layer. If there is only one stroke appearing in the units of a given quantum, the minimum expected risk is calculated as $\rho_{ij}{}^5$, as shown in (11). Since there are four units in each quantum and only one stroke in the quantum, the total probability is the sum of probabilities of strokes appearing in the four units, as shown in (12). If two strokes appear, each in one of two quanta and the appearance of the strokes is dependent, the minimum expected risk is the product of the probabilities of the two quanta, namely, $\rho_{ij}{}^4$ in (13). If three strokes appear, each in one of three quanta, the minimum expected risk $\rho_{ij}{}^3$ is expressed in (14). If four strokes appear, each in one of four quanta, we can calculate $\rho_{ij}{}^2$ as in (15). Also, if five strokes appear, each in one of five quanta, we can express $\rho_{ij}{}^1$ as in (16).

There are five quanta in each layer, so that we can obtain five sequential equations. The first stage, second stage, and the fifth stage are expressed in (17), (18), and (19). If there are 50 quanta and 100 units in each layer, the number of sequential equations is equal to 50.

The algorithm used in this paper is different from other algorithms. It is a special operating principle for recognizing (two-dimensional) block pictures. Also we can recognize three-dimensional objects with XYZ coordinates. This paper uses specialized mathematics, but it is quite different from [3] and [4].

II. Mathematical Model

A. Definition of a Quadrix

The definition of quadrix in this paper is different from that of a matrix. A quadrix is a block sum rectangle as in (1) and (3). It is especially convenient for calculating the risk. The method of calculation is given in the Appendix. Let ${}_mP_{ij} = P_m(i,j)$ be the probability in the jth quantum of the ith layer for class m. The transition probability quadrix is as follows:

$$\begin{vmatrix} {}_mP_{11} & {}_mP_{12} & \cdots & {}_mP_{1j} & \cdots & {}_mP_{1r} \\ {}_mP_{21} & {}_mP_{22} & \cdots & {}_mP_{2j} & \cdots & {}_mP_{2r} \\ \vdots & \vdots & & \vdots & & \vdots \\ {}_mP_{i1} & {}_mP_{i2} & \cdots & {}_mP_{ij} & \cdots & {}_mP_{ir} \\ \vdots & \vdots & & \vdots & & \vdots \\ {}_mP_{s1} & {}_mP_{s2} & \cdots & {}_mP_{sj} & \cdots & {}_mP_{sr} \end{vmatrix} \tag{1}$$

where

$i = 1,2,\cdots,s$ (number of layers)
$j = 1,2,\cdots,r$ (number of quanta)
$m = 1,2,\cdots,M$ (number of class)

Manuscript received January 30, 1974; revised May 25, 1974.
The author is with the Department of Electrical Engineering, National Taiwan University, Taipei, Taiwan, Republic of China.