A METHOD FOR RECOGNITION OF HAND PRINTED CHARACTERS

by

JO-SUNG TSAI

Submitted to the Department of Electrical Engineering in partial fulfilment of the requirements for the degree of Master of Science

Department of Electrical Engineering

Faculty of Pure and Applied Science

The University of Ottawa

Ottawa, Canada

1968

Approved for the Department
of Electrical Engineering

Supervisor

Chairman of the Examining
Committee

Chairman of the Department

ACKNOWLEDGEMENTS

The author wishes to express his sincere thanks to his supervisor, Professor C. L. Sheng, for his generous encouragement, guidance and supervision that made possible the present thesis.

ABSTRACT

A practical and effective preprocessing and recognition method for the pattern recognition machine is presented. The pattern recognition machine considered is simulated on a digital computer. Samples of five pattern classes which consist of hand-printed characters, A, B, C, D, E, are used in the experiments. The results are found very satisfactory.

A preprocessing method, called Sequential and Centroid Transformation (SCT), is developed to extract the characteristic features from patterns. A recognition method, called Least Squares Recognition Algorithm, based on the properties of the SCT method is developed and is shown to satisfy the principle of least squares. One hundred samples which are collected randomly from several persons are used as input testing patterns. The samples used are not only translated, rotated, and dilated or shrunk but also found to contain a reasonable amount of random transformations such as change of proportions, bending of lines, etc. Nevertheless, the recognition rate is as high as 95 per cent

TABLE OF CONTENTS

CHAPTER 1

INTRODUCTION

1.1 General

The subject of pattern recognition is one aspect of artificial intelligence research. The behavior of the pattern recognition machine is essentially a decision making process in which input pattern is recognized as belonging to one of a finite number of pattern classes which are already learned through training process by the machine. The recognition process used by the machine can be viewed as a comparison of features of input pattern with the already known common features of the reference patterns. These machines can be used for character or speech recognition, weather forcasting, medical diagnoses, economic predictions and other related system tasks when the state of the system is represented as a pattern.

The structure of the pattern recognition machine considered in this work consists of two submachines, a receptor and a categorizer[1]. The receptor is a device which extracts the characteristic features from patterns. Hence, its input is a pattern to be recognized and its output is a set, X_1, X_2, ..., X_m, of measurements which characterize the pattern. This set of measurements will be designated by the vector X. The categorizer has the output, X, of the receptor as its input and assigns each of its input to one of a finite number, n, of categories or classes. Thus, the output of the categorizer is an integer i=1, 2, ..., n, where output i means that the categorizer has assigned its input to the ith category among the set of categories. Basically, the receptor is a preprocessing device which maps all possible input patterns in the pattern space or input space into feature space or property space; the categorizer is a classification device which maps the patterns in the feature space into output space.

In general, the pattern space can be considered as a function space. Any pattern can be described by a function defined on some given interval. This pattern function may be continuous or discrete depending on the nature of the pattern. It may be randomly distributed in the pattern space. Therefore, it may be a deterministic function or a random function. In the latter case, a pattern class corresponds to a stochastic process subject to a random environment (due to noise, distortion, etc.). In the former case, the rule of the variations of the pattern in one pattern class can be found or defined.

From another point of view, the variations of one pattern in the pattern space can be looked upon as transformations into pattern space itself. These variations of the pattern constitute a pattern class. When such transformations are known a priori, the pattern function is called a deterministic function. When they are unknown a priori, the pattern function is called a random function. Most of the pattern functions are complex ones. As a matter of fact, it is an important and fundamental procedure of pattern recognition to define a class of patterns after defining transformations on these patterns, which preserve the structure of these patterns. Therefore, it is often preferable to investigate the effects of these transformations on the feature space. In other words, it is desired to transform the transformations in the pattern space into the feature space, and see what transformations are eliminated or distorted and what transformations are preserved.

Consequently, the pattern recognition machine does not need to observe the whole pattern function. Instead, it needs only to sense the features of the pattern function from a set of measurements taken in the pattern space. Thus, these sets of measurements form a feature space or property space. Every pattern in the pattern space is mapped into a curve in the feature space and every pattern class is mapped into a set of curves or some set of curves. The function of the receptor is nothing but to carry out such a transformation so that these sets of curves of different pattern classes are separable in the feature space. When the receptor performs its transformation function, it shrinks the pattern space into feature space so that it can select the absolute features of the patterns for the categorizer. Unfortunately, there exists no general method and this subject is not much explored. It has, therefore, been the purpose of this work to develop a description of patterns which is completely independent of the position and size variation of a pattern in a field of observation

In the next section, some previous results of the preprocessing method are introduced

In Chapter 2 a preprocessing method called the Sequential Centroid Transformation (SCT) Method, which is the revised version of a report "A Preprocessing Method to Identify Pattern Classes with A Deterministic Pattern Function," [2], is presented.

The method consists of the following steps: first, finding the centroid of the pattern in the quantized pattern space; secondly, measuring the distances between the centroid and the contours of the pattern according to a certain fixed sequence for fixed angle increments; thirdly, taking logarithm of these distances and plotting them vs. angle on rectangular coordinates. In this way, every pattern class is mapped into a set of quantized curves. The properties of the quantized curves are studied. It is found that by proper operation it is possible to eliminate completely some effects such as translation, rotation, and size variation due to the deterministic transformations of the pattern.

The method applied to the two dimensional pattern space is in fact a conformal mapping and is justified by a theorem which is proved from a conformal mapping point of view.

In Chapter 3 an algorithm for the pattern recognition machine developed from the preceding results for the pattern recognition machine is presented. In that chapter it is also shown that the algorithm satisfies the principle of least squares. The algorithm is essentially to compare the quantized curves of the testing pattern and the quantized curves of the reference patterns to see to which of the latter curves the former curve is the closest. This is achieved by shifting horizontally the quantized curve of the testing pattern with respect to the quantized curve of each reference pattern until a minimum sum of the squares of the errors is found. Starting at these new positions, the quantized curves of the testing pattern is then shifted vertically with respect to each corresponding reference pattern until a final minimum sum of the squares of the errors is found. The testing pattern is then assigned to the class of a reference pattern which gives the minimum minimum sum of the squares of the errors. The method is straightforward and needs no cut and try at all during the searching minimum process.

The pattern recognition machine under consideration is programmed on the IBM 360 digital computer. One hundred samples which are collected randomly from several persons are used as input testing patterns. In presenting the character samples for recognition, eight measurements are made on each sample. The samples used are not only translated, rotated, and dilated or shrunk but also found to contain a reasonable amount of random transformations such as change of proportions, bending of lines, etc. Nevertheless, the experimental results are found very satisfactory. The recognition rate is as high as 95 per cent. If more measurements were made on each sample, the recognition rate could be improved further.

In Chapter 4 some concluding remarks are given.

1.2 Summary of Some Previous Results of Preprocessing Methods

During the recent years, considerable progress has been made in this new and exciting field. However, the increasing number of papers published shows that the scope and depth of this developing field has not yet been fully studied. For a unified and detailed treatment of major concepts and mathematical results underlying trainable devices for pattern recognition, the reader is referred to [3] and [4].

One of the most important problems encountered in pattern recognition has been the extraction of characteristic features from patterns. This is the subject of measurement selection, sometimes called preprocessing. Several preprocessing techniques are discussed in the literature for pattern recognition and related fields.

In reference [5] some descriptive processes including fields of thermodynamics, photography, language, models, gambling, cryptology, and pattern recognition are presented. It is shown that many descriptive processes can be reduced to mathematical terminology by dividing the domain of description into cells whose size is determined by the resolution, and filling the cells with occupants where the number of kinds of occupants is determined by the accuracy of distinguishing between them.

In reference [6] a preprocessing technique using pairs of elements which are independent of the position of the character is described. The basic idea underlying the techniques is that of registration invariance as called by the author. The starting point for this mode of description is to place a discrete mesh over the character and to set each mesh point to unity if it covers a black part of the character and to zero if it covers white. The patterns are then represented by counting specified pairs of ones regardless of their positions in the matrix for all possible kinds of pairs. However, the patterns are only described in such a way that shifting translationally in any direction does not affect any observed parameter.

In reference [7] the problem and the requirement of selecting m out of the M characteristics to be used by the recognition system is discussed where $m < M$. An experimental selection of these characteristics is required because the system designer, while able to conceive of a large number of relevant characteristics, does not know which are the most important. A statistical model of the system is then proposed. A set of statistics relating the patterns to the characteristics is also proposed to be calculated in order to select the best characteristics. To describe the characteristics, a horizontal line is moved down the grid from top to bottom. At each position of the line, the number of intersections with the letter is recorded; if the same number of intersections is observed for two or more consecutive positions of the line, then the number is only recorded once. The result of this procedure is a set of digits. The following four horizontal transition characteristics are then described: (1) the total number of digits,(2) value of the maximum digit,(3) whether the sum of the first n/2 digits is greater than, equal to, or less than the sum of the last n/2 digits, and (4) whether the sequence of digits is symmetrical about its center. Similarly, by scanning a vertical line from left to right four vertical transition characteristics are obtained. The pattern is enclosed by a box which is then sub-divided into sixteen equal sub-boxes. The number of black squares in each sub-box is then counted and several characteristics are selected. Yet the effectiveness of these characteristics is not guaranteed and so a statistical selection is required.

In reference [8] a method to measure the effectiveness of the receptors' output is discussed. The effectiveness is interpreted from a statistical point of view. By thinking of each category as generating a certain density of points at each location in the p-dimensional space where p is the number of measurements performed by the receptor, or, in other words, by thinking of the multivariate probability density function corresponding to each category. Now, if the mapping generated by the receptor is such that, at each location in the space, the density due to category i is equal to the density due to category j, then no categorizer will be able to discriminate between these two categories. The receptor that generates this mapping is then ineffective. Otherwise, it is effective. Roughly speaking, the ability of the receptor to discriminate between two given categories depends upon the 'distance' between the densities. How the effectiveness depends on the properties of the individual measurements and their interrelations are then presented. However, the problem of how an effective receptor actually may be discovered is not discussed in that paper.

In reference [9] an adaptive logic element called ADALINE and a parallel network of ADALINES called MADALINE are used as a pattern recognition machine. The characteristics selection procedure becomes the training procedure in this machine. The insensitivities of the pattern's rotation, translation and size can be trained into the machine with the minimum mean-square error procedure. This process is sometimes called the generalization of the pattern. After training, the machine can recognize new patterns never seen before which are quite unrelated to the training pattern set. Unfortunately, owing to the capacity and slow training procedure of the machine, the kinds of insensitivities that may be trained-in are restricted. For example, rotation can not be trained-in in all orientations. In that paper, only four rotations are trained-in, that is, rotation of a pattern by 90 degrees each time. Besides, by using a single training routine, a wide variety of learning and generalization processes can be induced merely by designing appropriate sets of training patterns.

In reference [10] an automatic construction of recognition measurements based on the analysis of representative data is developed. Reference [11] and reference [12] are the extension of the method described in reference [10]. Starting from some typical samples of the patterns to be recognized, the designing process of the so called character recognition logics which are used to represent the effective measurements of the characters is summarized as follows: Recognition logics consist of only certain types of N-tuples. Each N-tuple consists of five to nine points in a prescribed spatial arrangement, and with a prescribed assignment of black and white states for each point. These N-tuples are shifted with respect to the input character so that the N-tuples are tested for a match in all possible positions. If a certain N-tuple matches a character in any position, the output of that measurement is "ONE." A large set of N-tuples is thus designed at first, say M in number. From these M measurements, a minimal subset of L in number is selected by a minimum distance decision procedure [10]. So far, the particular recognition logics discussed in those papers are chosen on the basis of being invariant under translation only as long as the character remains entirely within the field of observation. Besides, the size can only differ by 15 per cent from a standard size. Rotation invariance is not considered at all.

In reference [13] a technique to examine certain k-tuples of pattern points which are selected randomly and disjointly from N binary pattern points contained in a two dimensional pattern and to attempt to separate patterns on the basis of these subpatterns was viewed from a statistical approximation sense. Again, the effectiveness of this technique is not guaranteed.

In reference [14] a procedure to abstract the line segments of the line patterns was presented. Each segment's length and angle, and its bearing direction and distance from the pattern's centroid are recorded in a matrix form. Each line segment is represented by a one in the matrix. The columns of this matrix represent the bearing direction of segment's midpoint from the pattern's centroid. The matrix's rows represent the segment's angle with respect to the assumed horizontal reference. Each element of the matrix is then subdivided into a three by three submatrix. The segment lengths are quantized into three sizes and are represented as the rows of the submatrix. The columns of the submatrix represents the quantized distance separating the centroid from the segment's midpoint. The entries are again zero and one. However, the effectiveness of this representation is not justified and the representation is very complicated.

In reference [15] a two dimensional binary pattern function is transformed int certain combinations of normalized moments which are invariant to the location, size, stretching and squeezing in either vertical or horizontal directions, and rotation through a small angle of the pattern. Yet the selection and computation of the moments are very complicated, e.g. selection of the moments is achieved by trial and error method.

In reference [16] the autocorrelation followed by Fourier analysis of a normalized radial scan of the pattern was proposed for position and size invariant two-dimensional pattern recognition. The two-dimensional spatial frequency spectrum and its Fourier transform, the two-dimensional autocorrelation function, are independent of translations of the original pattern just as the autocorrelation and spectrum of an electrical signal are independent of time delays of the original waveform. The effect of rotating input pattern around any point is merely to rotate the corresponding autocorrelation function about the corresponding point. In order to abstract the rotation invariance the autocorrelation is expressed in terms of polar coordiantes. Three method to eliminate the size invariance have been proposed. One is to average out the radial variations of the autocorrelation. The second one is to retain the information associated with the radial variation of the autocorrelation. The third one is to form the appropriate autocorrelation function corresponding to scale changes of the original (and hence the autocorrelation) pattern.

Reference [17] is related to reference [15] and reference [16] except that the computations are simpler. The two-dimensional binary pattern is Fourier transformed to eliminate the translation effect of the pattern. By conformal mapping and vertical toroidalization, the rotations and dilations of the Fourier transformed pattern, in the polar coordinates, are mapped into rectangular coordinates again where rotations and dilations of the Fourier transformed pattern become translations in horizontal and vertical directions. Since the transformed pattern in the rectangular coordinates is a periodic function, Fourier series is used to transform this pattern to eliminate the rotation and dilation effects. Owing to the multi-stage transformations and approximations, the information of the features is dissipated. Therefore, the recognition rate is not very good in testing experiment for hand-printed characters as indicated in that paper.

CHAPTER 2

A PREPROCESSING METHOD

2.1 The Sequential Centroid Transformation (SCT) Method

Consider a two dimensional pattern space. The procedure of the Sequential Centroid Transformation (SCT) consists of three steps. First, the center of gravity of the pattern is found by the following well-known equations:

$$\bar{X} = 1/M \int\int_R x f(x, y)\, ds$$
$$\bar{Y} = 1/M \int\int_R y f(x, y)\, ds$$
$$M = \int\int_R f(x, y)\, ds.$$

Secondly, the center of gravity of the pattern is considered as the origin of the polar coordinates. The patterns are then transformed into rectangular coordinate plots. Lastly, the transformation curves in the rectangular coordinates are mapped into logarithmic transformation curves. In other words, the ordinate is scaled proportional to the logarithm of the transformation curves. Mathematically, the preceding pattern transformation technique can be summarized as following:

$$f(x, y) \longrightarrow T_1 f(x, y) \longrightarrow T_2 T_1 f(x, y)$$

where

$f(x, y)$ denotes the original input pattern in the rectangular coordinates.

T_1 denotes the transformation from rectangular coordinates into polar coordinates.

T_2 denotes the transformation from polar coordinates into logarithmic rectangular coordinates.

Unfortunately, it is completely impractical and difficult to obtain the centroid of a complicated pattern in the continuous case. That is to say, for some patterns, the function $f(x, y)$ can not be found. Therefore, only the discrete case is considered in this work.

Furthermore, a descriptive process can be reduced to mathematical terminology by dividing the domain of description; i.e., pattern space, into cells whose size is determined by the resolution. The cells are then filled with occupants. Therefore, a two-dimensional pattern space of black and white can be represented to any degree of resolution by a M x N matrix of zeros and ones. The ones correspond to cells which are in the black portions of the pattern, and the zeros to cells in the white.

2.2 Algorithm of the SCT method.

(1) First, the centroid of the pattern is found. For simplicity of illustration, the pattern space is divided into an array M x N of cells, say, M = 20, N = 20. Since it is a two dimensional

pattern, two characteristics, namely, the X and Y positions are enough to distinguish the cells from one another. The centroid of the pattern can be calculated by the following equations:

$$\bar{X} = 1/S \sum_{j=1}^{M} \sum_{k=1}^{N} x_j m_{jk}$$

$$\bar{Y} = 1/S \sum_{j=1}^{M} \sum_{k=1}^{N} y_k m_{jk}$$

$$S = \sum_{j=1}^{M} \sum_{k=1}^{N} m_{jk} ,$$

where $m_{jk} = 0$, if the cell is not occupied, i.e., white;

$m_{jk} = 1$, if the cell is occupied, i.e., black.

(2) The centroid of the pattern is set as the origin of the polar coordinates. Starting with a scanning vector, the angles $\{\theta_i\}$ and the radii $\{r_i(j)\}$ along the contours of the pattern are measured. The measurements can be made at every $\Delta\theta$ radian. Thus, a set of data can be obtained for each pattern.

(3) The measured values of the angle and the radius are plotted as a rectangular coordinate graph. Therefore, a quantized transformation curve is obtained.

(4) The quantized transformation curve is mapped into quantized logarithmic transformation curve by the following equations:

$$X_{ji} = \log_e 10r_i(j), \qquad \text{if } r_i(j) > 0;$$

$$X_{ji} = 0, \qquad \text{if } r_i(j) = 0.$$

Where j denote the multiplicity of the variable θ_i ,

i denote the order of the measurement taken.

Example 1.

The pattern 'A' is shown in Figure 1.

Step 1. $\bar{X}$ = (5x1 + 6x1 + 7x2 + 8x2 + 9x3 + 10x3 + 11x3 + 12x2 + 13x3 + 14x5 +15x3 + 16x4)/32

= 11.6

$\bar{Y}$ = (5x1 + 6x2 + 7x2 + 8x3 + 9x4 + 10x6 + 11x3 + 12x4 + 13x2 + 14x2 + 15x3)/32

= 10.3

Step 2. The data set is shown in Table 1.

Step 3. The quantized transformation curve is shown in Figure 2.

Step 4. The preceding transformation curve is amplified by a factor '10'. The reasons can be seen in Section 2.3, Section 2.4, and Section 3.3. The values of X_{ji}, where $j = 1,2$, $i = 1,2, .., 8$, are listed in Table 2. The quantized logarithmic transformation curve is shown in Figure 3.

Table 1.

θ_i	0	$\pi/4$	$\pi/2$	$3\pi/4$	π	$5\pi/4$	$3\pi/2$	$7\pi/4$
$r_i(1)$	3.8	4.0	2.6	1.6	2.0	1.4	1.0	1.5
$r_i(2)$	0.0	0.0	0.0	0.0	0.0	0.0	0.0	7.4

Table 2.

X_{11}	X_{12}	X_{13}	X_{14}	X_{15}	X_{16}	X_{17}	X_{18}
3.6376	3.6889	3.2581	2.7726	2.9957	2.6391	2.3026	2.7080
X_{21}	X_{22}	X_{23}	X_{24}	X_{25}	X_{26}	X_{27}	X_{28}
0.0	0.0	0.0	0.0	0.0	0.0	0.0	4.3041

2.3 Properties of the SCT method

(1) The transformation curve is a periodic function with a period equal to 2π. This curve is independent of the translation effect of the pattern. In other words, when a pattern is mapped from polar plot into rectangular coordinate plot, SCT automatically takes care of the translation effect of the pattern.

(2) If one pattern is different from another one simply due to rotation around the centroid of the pattern or due to translation and rotation of the pattern, then it is easily checked by observing the transformation curves. Since the transformation

curves are periodic functions, with a period equal to 2π, the curves can be shifted horizontally to see if they are the same as other ones or not.

(3) The size-variants can be eliminated by finding the differences of the two logarithmic transformation curves. If the differences at all the quantizing points are the same, then it can be concluded that the two patterns belong to the same pattern class. This is similar to the concept of plotting the gain-phase diagram on the Nichols chart [18]. The variable gain loci have identical forms except for the shift in vertical position.

Mathematically, let $f(\theta)$ represent the transformation curve. Then, all the transformation curves with a form of $kf(\theta)$ belong to the same pattern class, where k is a constant.

In fact, by taking the logarithm of $kf(\theta)$, it has been found that the logarithmic transformation curves $\log_e kf(\theta)$ have the same waveshape as $\log_e f(\theta)$. This can be seen from the following equation:

$$\log_e kf(\theta) = \log_e k + \log_e f(\theta).$$

The difference between the two logarithmic transformation curves is $\log_e k$, which is independent of the variable θ. Therefore, it can be concluded that the two patterns belong to the same pattern class if their logarithmic transformation curves are the same except for a shift in vertical position.

(4) Some patterns which belong to different pattern classes only because of their different position or size may have the same transformation curve. In such case, these patterns can not be distinguished.

(5) The transformation curve may be a multivalued function. However, this does not alter the topological relationship among the transformation curves which belong to the same pattern class.

(6) If one pattern is an image of another one with respect to a line in the pattern space, then the former's transformation curve is also an image with respect to the vertical axis in the feature space of the latter.

Example 2.

Property (1) and property (2) can be seen from Figure 4, Figure 5, and Figure 6. The two triangles in Figure 4 are in different positions due to translation and rotation of the pattern. Their logrithmic transformation curves are shown in Figure 5 and Figure 6 respectively. If the curve in Figure 6 is shifted left by ϕ, then the curves in Figure 5 and Figure 6 become the same, which means the two patterns belong to the same pattern class.

Property (3) is demonstrated in Figure 7 and Figure 8. In Figure 7, pattern (a) and pattern (b) belong to the same pattern class, because in Figure 8 the differences at all the quantizing points of the two logarithmic transformation curves are the same, which means one logarithmic transformation curve can be shifted upward or downward to superpose on the other one.

Property (4) is obvious. For instance, pattern '6' and '9' can not be distinguished since their logarithmic transformation curves are identical. Property (5) is shown in Figure 9 and Figure 10. It was found that after the necessary shifting, horizontal and vertical, of one curve (small 'A') with respect to the other one (large 'A'), the two curves will coincide with each other. It is then concluded that the two patterns in Figure 9 belong to the same pattern class irrespective of their position, size, and multiplicity of the measured values.

Property (6) is clearly shown in Figure 11 and Figure 12.

2.4 Exposition of the Sequential Centroid Transformation.

In order to dig more deeply into the problem, it is useful to analyze the Sequential Centroid Transformation from a conformal mapping viewpoint. Let P represent the two-dimensional pattern space; F represent the two-dimensional feature space. Then, the Sequential Centroid Transformation described in the preceding section is simply to find the centroid of the pattern in the pattern space, and to measure the X_1, X_2, .., X_m, totally m distances between the centroid and the boundary points according to a fixed finite sequence for fixed angle increments. These distances are then transformed into logarithmic form, $\log_e X_1$, $\log_e X_2$, ..., $\log_e X_m$. The centroids of the different patterns are mapped into the abscissa of the F-space. The pattern itself is mapped into a curve in the F-space. Mathematically, the preceding procedure can be expressed as follows: Let $z = e^{j\theta}$ represent the scanning vector starting from the centroid of the pattern in the P-space where r and θ are variables, After transformation it becomes $\log_e z = \log_e r + j\theta$ where again r and θ are variables in the F-space. The abscissa is scaled proportional to the $j\theta$; the ordinate is in proportion to the $\log_e r$.

In order to investigate the effects of this transformation, it is necessary to define some terms to be used hereinunder.

Definition 1. A pattern class is defined as the set of patterns resulting from the transformations of a reference pattern into the pattern space itself such that the structure of the reference pattern is preserved.

Definition 2. An Euclidean transformation (or isometry) of a space into itself is a transformation $T \rightarrow T'$ which preserves distances. Using the notation $T' = f(T)$ the defining property of a Euclidean transformation is

$$d(A, B) = d(f(A), f(B))$$

where $d(A, B)$ is the distance between A and B.

Definition 3. A similarity transformation of a space into itself is a transformation $T \rightarrow T'$ which preserves the ratio of distances. In this case distances are changed but uniformly magnified. This may be achieved by a congruence (Euclidean transformation) followed by a magnification.

Definition 4. Squashing of a space into itself is a transformation $T \rightarrow T'$ which maps lines to lines, and mid-point to mid-point. Yet distances, angles, and ratios of distances change. Areas change by a constant factor k, and distances between two points change by a varying factor g depending on the orientation of the line joining the two points. For instance, in Figure 13, the factor g varies from a 1 for a segment parallel to L, to 1/2 for a segment perpendicular to L. Such transformations preserve the following equation:

$$f(aA + bB) = af(A) + bf(B).$$

where $a+b = 1$, A and B represent two points in the space.

Theorem 1. Under the Sequential Centroid Transformation, the similarity transformations of the pattern in the P-space become the translations of transformation curves in F-space.

Proof: A similarity transformation is a congruence followed by a magnification.

According to the definition of the SCT the centroids of the patterns in the P-space are all sent into the abscissa of the F-space. In other words, the centroids of the patterns in the P-space are mapped into the same image in the F-space. Since the centroid represents the position of the pattern in the P-space, it is concluded that the position variant of the patterns is eliminated in the F-space.

Let

$$z = re^{j\theta}$$

represent the scanning vector in the P-space, then after SCT

$$\log_e z = \log_e r + j\theta \quad .$$

Let

$$u = se^{j\phi}$$

represent a constant vector, then 'uz' can be interpreted as rotating each 'z' through a fixed angle equal to ϕ and then multiplying its length by the factor 's'. This is to say, the pattern is rotated through an angle equal to ϕ and then magnified by a factor 's'. After SCT

$$\log_e uz = \log_e s + \log_e r + j(\theta + \phi)$$

and the transformation curve again have the same waveshape except for the shift in horizontal and vertical positions since $\log_e s$ and ϕ are constant. Q.E.D.

In general, the problem of pattern recognition can be considered as a recognition of stochastic process with each process representing a pattern class. Thus, the transformation curves, $f(\theta)$, of a certain pattern class represent a stochastic process where for any θ, $f(\theta)$ is a random variable. In other words, $f(\theta)$ is a family of θ-functions depending on the variations of a reference pattern in the pattern space, for example, squashing of a pattern in the pattern space, noise, deviation of centroid, etc. The reason of misrecognition is largely due to lack of a priori knowledge of the measurement data. Some existing method treated the position and size variants from a probabilistic point of view and error is an inherent property of such a treatment because of the fact that the probability density function is unknown. The SCT method treats this problem from a deterministic point of view and completely eliminates them. In this way, the uncertainty about the $f(\theta)$ is largely reduced.

CHAPTER 3

A RECOGNITION METHOD

3.1 General Machine Description

Based on the preceding analysis, a systematic and straightforward recognition algorithm has been developed for the pattern recognition machine. Owing to the characteristics of the Sequential Centroid Transformation Method, the pattern to be recognized by the machine under consideration are restricted to line patterns. Furthermore, hand-printed English letters were chosen as the input patterns for detailed simulation of the proposed pattern recognition machine on a general purpose digital computer.

Before digging more in detail, it is useful to have a look at the general machine implementation. The pattern recognition machine considered in this work consists of two submachines. The front-machine is a data preprocessing device called receptor which extracts the pattern's features according to the SCT method described in the preceding chapter. The tail-machine is a data classifying device called categorizer which tries to classify the transformed pattern into one of a finite number of classes. The algorithm used by the categorizer is to be developed in this chapter.

In order to mechanize the function of the receptor, a computer controlled CRT scanner is proposed for this purpose [Fig. 14]. A CRT scanner performs two functions on each pattern. The first function is a horizontal or vertical scanning of the pattern. The resolution ability of the scanner may be a 20 by 20 quantized point plot of the pattern field. With the horizontal and vertical axis placed along the bottom and leftmost edge of the pattern field, each quantized point is assigned a set of rectangular coordinate value. Each black point is weighted by a value 1 and white point 0. After scanning, the computer calculates and locates the centroid of the pattern according to the equations in Step 1 of Section 2.2. The second function is a rotational scanning along the contours of the pattern. With a counter-clockwise rotational scanning vector starting at the centroid of the pattern and ending at the edges of the pattern field, the values of the radius and the angle of the scanning vector with respect to a horizontal reference are measured. The measurements are recorded at the instant when the scanning vector encounters the contours of the pattern. The measurements are then mapped into logarithmic scale by the equations in Step 4 of Section 2.2. Another possible approach to this problem is that after horizontal or vertical scanning, the computer memorizes the whole quantized pattern in its memory and performs the measurement function by a stored program which calculates and locates the centroid and measures the distances between the centroid and quantized contours of the pattern.

In this work, however, the measurement function of the receptor is performed manually with the aid of a desk calculator provided in the computer center at the University of Ottawa. The logarithm function of the receptor is implemented within the recognition program described in Section 3.3.

The recognition algorithm utilized by the categorizer is developed from the results of the preceding chapter. Essentially, an unknown input pattern vector, $X(x_1, x_2, .., x_m)$, is compared with a set of stored reference pattern vectors, $Y_1(y_1, y_2, .., y_m)$, $Y_2(y_1, y_2, .., y_m)$, .., $Y_n(y_1, y_2, .., y_m)$. Since the feature vectors, X and Y_r, where r=1, 2, .., n, are periodic, , X is first shifted horizontally with respect to each Y_r during the comparisons. In other words, since X has m components, $x_1, x_2, .., x_m$, m comparisons are made with each reference pattern vector, Y_r, by shifting X m times. Then, the image of the input pattern vector X is formed and the same comparisons are made again. Thus, 2m horizontal error vectors, $Z_{pr}(z_1, z_2, .., z_m)$, where p=1, 2, .., 2m, are generated with each reference pattern vector, Y_r, according to the following equation:

$$Z_{pr} = X_p - Y_r$$

where p denotes the horizontal shift position of the input pattern vector, X, that is, p, from 1 to m, denotes the horizontal shift position of the original X and, from m+1 to 2m, denotes the horizontal shift position of the image of the original X.

The sum of the squares of the errors is then calculated for each Z_{pr} by the following equation:

$$Q_{pr} = \sum_{i=1}^{m} z_i^2 \quad .$$

The minimum Q_{pr} for each r is selected and the corresponding Z_{pr} and X_p are identified. The sum of the errors,

$$S_{pr} = \sum_{i=1}^{m} z_i \quad ,$$

is also calculated for the identified Z_{pr} corresponding to the minimum Q_{pr}. The identified X_p is then shifted upward or downward depending on whether S_{pr} is negative or positive to make S_{pr} equal to zero or approach zero. With X_p in this new position, the corresponding Q_{pr} is calculated again and is a minimum as shown in the next section.

Thus, after horizontal shift and vertical shift of X with respect to each Y_r, a minimum Q_{pr} is determined. Totally, there are n minimum Q_{pr} since there are n reference pattern vectors, $\{Y_r\}$. From these n minimum Q_{pr}, denoted by min Q_{pr}, minimum min Q_{pr} is found and the corresponding Y_r is identified. The input pattern vector is then assigned to the corresponding pattern class of Y_r. This decision procedure can be called a Least Squares Recognition Algorithm.

3.2 Justification of the Recognition Machine's Performance

The purpose of the decision procedure described in the preceding section is to select, from 2mn comparisons, the class of the reference pattern vector, Y_r, which gives the minimum sum of the squares of the errors, as the most probable pattern class for the input pattern vector, X.

In order to justify this algorithm, it is necessary to express the Z_{pr} in terms of its elements. The 2mn error vectors are shown in the following:

$$\begin{matrix} Z_{11} & Z_{12} & \cdots & Z_{1n} \\ Z_{21} & Z_{22} & \cdots & Z_{2n} \\ \cdots & \cdots & \cdots & \cdots \\ Z_{(2m)1} & Z_{(2m)2} & \cdots & Z_{(2m)n} \end{matrix} \quad .$$

Moreover,

$$z_j = x_i - y_j,$$

where $i = 1, 2, \ldots, m$; $j = 1, 2, \ldots, m$.

For any reference pattern vector, $Y_r(y_1, y_2, \ldots, y_m)$, there are 2m error vectors:

$$Z_{1r} = \{z_i | z_1 = x_1 - y_1, z_2 = x_2 - y_2, \ldots, z_m = x_m - y_m\}$$

$$Z_{2r} = \{z_i | z_1 = x_2 - y_1, z_2 = x_3 - y_2, \ldots, z_m = x_1 - y_m\}$$

$$\cdot\cdot \quad \cdot\cdot \quad \cdot\cdot \quad \cdot\cdot \quad \cdot\cdot$$

$$Z_{mr} = \{z_i | z_1 = x_m - y_1, z_2 = x_1 - y_2, \ldots, z_m = x_{m-1} - y_m\}$$

$$Z_{(m+1)r} = \{z_i | z_1 = x_m - y_1, z_2 = x_{m-1} - y_2, \ldots, z_m = x_1 - y_m\}$$

$$Z_{(m+2)r} = \{z_i | z_1 = x_{m-1} - y_1, z_2 = x_{m-2} - y_2, \ldots, z_m = x_m - y_m\}$$

$$\cdot\cdot \quad \cdot\cdot \quad \cdot\cdot \quad \cdot\cdot \quad \cdot\cdot$$

$$Z_{(2m)r} = \{z_i | z_1 = x_1 - y_1, z_2 = x_m - y_2, \ldots, z_m = x_2 - y_m\}$$

According to the principle of least squares [19], which says that the most probable pattern class of the input pattern vector X obtainable from a set of comparisons with respect to the stored reference pattern vectors, $Y_1, Y_2, \ldots, Y_n$, is that class of the reference pattern vectors, Y_r, for which the sum of the squares of the errors is a minimum. Now, by the assumption that all the m separate measurements are independent events the following lemmas and theorem can be derived.

<u>Lemma 1</u>: $\mathcal{S}_{pr} = z_1 + z_2 + .. + z_m$ is a constant with respect to the subscript variable, p, that is, $\mathcal{S}_{pr}$ is a constant for each set of 2m error vectors, $z_{1r}, z_{2r}, \ldots, z_{mr}, z_{(m+1)r}, z_{(m+2)r}, \ldots, z_{(2m)r}$.

Proof: $$\mathcal{S}_{pr} = \sum_{i=1}^{m} z_i$$

$$= \sum_{i=1}^{m} (x_q - y_i)$$

Where $q = p + i - 1, \quad q \leq m$

$q = i + p - m - 1, \quad q > m$

That is, $$\mathcal{S}_{pr} = \sum_{q=1}^{m} x_q - \sum_{i=1}^{m} y_i$$

$$= \sum_{i=1}^{m} x_i - \sum_{i=1}^{m} y_i$$

Therefore, $\mathcal{S}_{pr}$ is independent of p .

Q.E.D.

<u>Lemma 2</u>: In the ideal case, subject to the constraint

$$S_{pr} = k = \text{constant}$$

with respect to the subscript variable, p, the minimum of

$$Q_{pr} = z_1^2 + z_2^2 + .. + z_m^2$$

occurs when

$$z_1 = z_2 = .. = z_m$$

Proof: By the usual Lagnangian multiplier technique, it is obtained that

$$Q_{pr} + \lambda S_{pr} = z_1^2 + z_2^2 + .. + z_m^2 + \lambda(z_1 + z_2 + .. + z_m - k)$$

Furthermore, the following m+1 equations can be obtained:

$$2z_1 + \lambda = 0$$

$$2z_2 + \lambda = 0$$

$$\vdots$$

$$2z_m + \lambda = 0$$

$$S_{pr} = k$$

By adding the first m and using the last equation,

$$\lambda = -\frac{2k}{m}$$

is found.

Using this relation,

$$z_1 = z_2 = .. = z_m = \frac{k}{m}$$

Q.E.D.

This lemma justifies that by selecting the horizontal minimum Q_{pr}, the input pattern vector, X, has to have, under the ideal situation, the same waveshape as the corresponding reference pattern vector, Y_r, if X is to be assigned to the pattern class of Y_r. In fact, it is this lemma which justifies that the pattern recognition machine under consideration has the ability to eliminate the rotation and image effects (Euclidean transformation) of the pattern.

Lemma 3: It is always possible to shift vertically up or down any input pattern vector, X, an amount S_{pr}/m for each x_i's with respect to any reference pattern vector, Y_r, such that

$$Q_{pr} = z_1^2 + z_2^2 + .. + z_m^2$$

has one and only one vertical minimum point.

Proof: Since

$$S_{pr} = z_1 + z_2 + .. + z_m, \text{ rearrange the items,}$$

$$(z_1 - S_{pr}/m) + (z_2 - S_{pr}/m) + .. + (z_m - S_{pr}/m) = 0.$$

This always can be done by shifting the input pattern vector, X, vertically, up or down depending on whether S_{pr} is negative or positive, an amount S_{pr}/m. Let

$$Q_{pr} = z_1^2 + z_2^2 + .. + z_m^2 \quad ,$$

and after vertical shift of X

$$\text{new } Q_{pr} = (z_1 - S_{pr}/m)^2 + (z_2 - S_{pr}/m)^2 + .. + (z_m - S_{pr}/m)^2$$

$$= z_1^2 + z_2^2 + .. + z_m^2 + mS_{pr}^2/m^2 - (2S_{pr}/m)(z_1 + z_2 + .. + z_m)$$

$$= Q_{pr} + S_{pr}^2/m - 2S_{pr}^2/m$$

$$= Q_{pr} - S_{pr}^2/m \ .$$

The last item is always positive since m is the number of measurements, which is always positive. Therefore,

$$\text{new } Q_{pr} < Q_{pr}, \text{ if } S_{pr} \neq 0;$$

$$\text{new } Q_{pr} = Q_{pr}, \text{ if } S_{pr} = 0 \ .$$

This means that by shifting the input pattern vector, X, vertically to make $S_{pr} = 0$, there exists one and only one vertical minimum point of Q_{pr} . Q.E.D.

This lemma justifies that the pattern recognition machine has the ability to eliminate the dilation or shrinking effect of the input pattern and to reduce the effort for searching the vertical minimum point to a single step.

Now, let

$$Q_{pr} = z_1^2 + z_2^2 + .. + z_m^2, \; p = 1, 2, .., 2m,$$

denote the sum of squares of errors in any set of 2m error vectors, $Z_{1r}, Z_{2r}, \ldots, Z_{(2m)r}$;

$$Q_{kr} = z_1'^2 + z_2'^2 + .. + z_m'^2$$

$$= Q_{pr}\Big|_{p=k}$$

$$= \text{minimum } Q_{pr}$$

denote the minimun sum of squares of errors;

$$Q_{jr} = Q_{pr}\Big|_{p \neq k} .$$

It is obvious that

$$Q_{jr} \geq Q_{kr} \text{ , for any j.}$$

Lemma 4: If $Q_{jr} \geq Q_{kr}$, then after vertical shift of X with respect to Y_r, new $Q_{jr} \geq$ new Q_{kr} .

Proof: By lemma 3 in this section, the amount of X to be shifted is S_{pr}/m, that is,

$$\text{new } Q_{jr} = (z_1 - S_{pr}/m)^2 + (z_2 - S_{pr}/m)^2 + \ldots + (z_m - S_{pr}/m)^2$$

and

$$\text{new } Q_{kr} = (z_1' - S_{kr}/m)^2 + (z_2' - S_{kr}/m)^2 + \ldots + (z_m' - S_{kr}/m)^2.$$

$$\begin{aligned}\text{new } Q_{jr} - \text{new } Q_{kr} = Q_{jr} - Q_{kr} &- 2S_{pr}/m\,(z_1 + z_2 + \ldots \\ &+ z_m) + 2S_{kr}/m\,(z_1' + z_2' + \ldots + z_m') \\ &- S_{pr}^2/m + S_{kr}^2/m\,.\end{aligned}$$

By lemma 1 of this section,

$$\begin{aligned}S_{pr} &= z_1 + z_2 + \ldots + z_m \\ &= z_1' + z_2' + \ldots + z_m'\,.\end{aligned}$$

Therefore, by the assumption $Q_{jr} \geq Q_{kr}$,

$$\begin{aligned}\text{new } Q_{jr} - \text{new } Q_{kr} &= Q_{jr} - Q_{kr} \\ &\geq 0.\end{aligned}$$

Q.E.D.

This lemma justifies that by searching firstly the horizontal minimum point of Q_{pr} and then vertical minimum point of Q_{pr} for each r the latter is the real minimum point.

<u>Theorem 1:</u> Under the Least Squares Recognition Algorithm, which compares the input pattern vector, X, with the stored reference pattern vector, Y_r, r = 1, 2, ..., n, by searching firstly the horizontal minimum point of Q_{pr} and then the vertical minimum point of Q_{pr}, the input pattern vector, X, is assigned to a pattern class of Y_r that gives the minimum minimum sum of the squares of the errors. This assignment is optimal in the sense of the least squares principle.

Proof: The theorem is obvious by lemma 2, lemma 3, and lemma 4. Q.E.D.

Corollary 1: The optimalities of the recognition algorithm apply to the multi-valued input pattern vector, X, i.e. X is a group of vectors in matrix form as can be seen in the example 1 of Section 2.2 instead of a single vector.

The theorem and corollary justify that the recognition algorithm conforms to the properties of the SCT method. Therefore, the pattern recognition machine is able to recognize the patterns which are under translation, rotation, and dilation. Furthermore, a reasonable amount of random transformations of the pattern to be recognized is also permitted. Note that there is a definite and straightforward method to search for the horizontal and vertical minimum points. Therefore, the process needs no cut and try at all.

However, in the practical application, lemma 3 needs some modification. Since X is a discrete set of points in the feature space and some of these points, which contain information of the empty contours of the pattern, have a value zero, it is obvious that when X is shifted vertically those points of X which have a value zero should be unchanged. This requirement induces the following situations.

The following equations:

$$S_{pr} = z_1 + z_2 + .. + z_m ,$$

and $$z_j = x_i - y_j, \text{ where } j = 1, 2, ..., m, \quad i = 1, 2, ..., m,$$

are defined in Section 3.1 and at the beginning of this section.

(1) $x_i \neq 0$, $y_j = 0$, or $y_j \neq 0$, hence $z_j \neq 0$, or $z_j = 0$. Let $S_{pr1} = z_1 + z_2 + .. + z_{m_1}$, where $m_1 < m$, represent this kind of errors. S_{pr1} can be reduced to zero by vertical shift of X.

(2) $x_i = 0$, but $y_j \neq 0$, hence $z_j \neq 0$. Let

$$S_{pr2} = z_{m_1+1} + z_{m_1+2} + .. + z_{m_2} , \text{ where } m_2 < m,$$

represent this kind of errors. S_{pr2} can not be changed by vertical shift of X.

(3) $x_i = 0$, and $y_j = 0$, hence $z_j = 0$. Let

$$S_{pr3} = z_{m_2+1} + z_{m_2+2} + .. + z_m$$

$$= 0,$$

represent this kind of error. S_{pr3} is always equal to zero during the vertical shift of the input pattern vector, X. The preceding situations are illustrated in Figure 15.

To sum up,

$$S_{pr} = S_{pr1} + S_{pr2} + S_{pr3} \; .$$

Now, since S_{pr2} can not be changed, its corresponding sum of squares of the errors, Q_{pr2}, can not be changed either. S_{pr3} is always equal to zero. Therefore, in order to minimize Q_{pr} by vertical shift, it is sufficient to make $S_{pr1} = 0$. This can be achieved by shifting the input pattern vector, X, vertically, up or down depending on S_{pr1} is negative or positive, an amount S_{pr1}/m_1 for each point of X which is not equal to zero. In this way, the sum of errors, S_{pr}, has been reduced to S_{pr2} which can not be reduced any more. The following equation demonstrates how the vertical shift of X can be done.

$$S_{pr2} = (z_1 - S_{pr1}/m_1) + (z_2 - S_{pr1}/m_1) + .. + (z_{m_1} - S_{pr1}/m_1) + z_{m_1+1} + z_{m_1+2} + .. + z_{m_2} + z_{m_2+1} + z_{m_2+2} + .. + z_m \; .$$

3.3 Machine's Realization on The Digital Computer

The pattern recognition machine was realized on the I. B. M. 360 digital computer located at the Computing Center, University of Ottawa, except that the measurement function of the receptor was performed manually as described in Section 3.1. While the techniques developed are applicable to generalized line patterns, the particular subset chosen for detailed test are hand-printed characters. The algorithms to be realized on the digital computer are essentially the same as described in Section 2.2 and Section 3.1 except a few modifications.

Actual samples of hand-printed characters, A, B, C, D, E, were used as input data for analysis. The samples used are not only allowed to be translated, rotated, dilated, and shrunk within a 20 by 20 squares field of pattern but also permitted a reasonable amount of random noise and certain other differences which appear as minor to a human reader, such as small change of proportions, slight bending of lines, etc. Some of the samples used are shown in Appendix 1.

The input data of each pattern, X or Y_r, is in the form of two row matrix the same form as Table 1 in Section 2.2 The components in the first row of X or Y_r denote the distance between the centroid and the first contour of the pattern encountered by the scanning vector in each direction. The components in the second row of X or Y_r denote the distance between the centroid and the second contour of the pattern encountered by the scanning vector in the corresponding direction. In this way, it is assumed that the pattern vectors, X and Y_r, are two-valued functions. This two-valued function assumption was found adequate for the samples used in the test. For more complicated patterns, n-valued function of X and Y_r can be used.

Since the values between $\log_e 0$ and $\log_e 1$ are negative, in order to simplify the computations negative values of X are to be avoided. Besides, the confusion between actual zero and $\log_e 1$ is also to be avoided. Therefore, all measurements taken are larger than the value 0.1 which can be estimated reasonably, the measurements are then multiplied by 10 and the logarithms of the measurements are calculated according to the equations in Step 4 of Section 2.2.

The input pattern vector, X, is then classified by its multiplicity. In other words, X is only compared with the reference pattern vector, Y_r, which has the same multiplicity as X. If the components in the second row of X are all equal to zero, then the multiplicity of X is 1; otherwise it is 2. The components in the first row of X can not be all equal to zero. Otherwise, it is a trivial pattern, that is, an empty pattern.

The comparisons between X and Y_r are then carried out according to the algorithm described in Section 3.1.

The program to implement the preceding algorithms is written in the Fortran IV language and is given in Appendix 2. It consists of a main program called "RECOGNITION PROGRAM' and three subprograms: "AMP", "LOGA", and "CALERR". The simplified flow chart of the main program is shown in Figure 16. The flow charts of the two subprograms: "AMP" and "LOGA" are not given because of their simplicity. The flow chart of the subprogram, "CALERR", is given in Figure 17.

The subprogram, "AMP", amplifies the pattern vectors, X and Y_r, by ten times to avoid the cases that $\log_e x_i \lesssim \log_e 1$ and $\log_e y_j \lesssim \log_e 1$ and to simplify the computations. The subprogram, "LOGA", computes the logarithm of X and Y_r according to the equations in Step 4 of Section 2.2 The subprogram, "CALERR", performs the horizontal shifting of X and storing of shifted X and calculates Z_{pr} and Q_{pr}.

In the main program, the pattern classes, A, B, C, D, E, are denoted as 1, 2, 3, 4, 5, respectively. The three subscripts arrays, X, Y, W, Z, SQZ, and V are used to denote X, Y_r, horizontal shift of X, Z_{pr}, squares of Z_{pr}, and image of X respectively. The first subscript of the preceding arrays denotes the columns of the data matrix and the second subscript denotes the rows of the data matrix. The third subscript in the arrays, X and Y, denotes the pattern classes. The third subscript in the arrays, W, Z, and SQZ, denotes the horizontal shift position of X, that is, the subscript variable, p, in Section 3.1 and Section 3.2. The third subscript in the array, V, is trivial. The one subscript arrays, SUMSQZ and SMALL, are used to denote Q_{pr} and min Q_{pr} respectively. The subscript in SUMSQZ denotes the subscript variable, p. The subscript in SMALL denotes the pattern classes. The scalars, SMAL, ERRNO, SUMZ, and SMIN, are used to denote the horizontal minimum Q_{pr}, m_1, S_{pr1}, and minimum min Q_{pr}. The actual class of the input pattern and the class, which is assigned by the pattern recognition machine, of the same pattern is printed out to show the effectiveness of the machine. The values of SMIN and SMALL are also printed out for heuristic study.

3.4 Experimental Results

The five different characters, A, B, C, D, E, out of hundred and five characters, which were collected randomly from several people by asking each person to print the letters within a two-inch 20 by 20 squares, were manually selected and identified as the reference patterns. The hundred characters were then used as testing patterns. The results of this run can be seen in Appendix 2 and are tabulated in Table 3.

Table 3. Results of testing 100 samples

		Input A	Input B	Input C	Input D	Input E
Output	A	19	0	0	0	0
	B	0	18	0	0	1
	C	0	0	19	0	0
	D	1	0	1	20	0
	E	0	2	0	0	19

It will be seen that the recognition rate was 95 per cent. While exact interpretation of each sample under test is difficult, an analysis of the results shows that the misrecognition of some samples such as A misrecognized as D are due to the coarse measurement of the patterns. This factor can be eliminated by increasing the number of measurements from 8 to, say, 16, or simply by rotating the horizontal reference scanning vector when taking the measurements. One extra advantage of forming image of X during the comparisons is to remove the error caused by the mislocation of the centroid in the symmetric pattern such as E.

CHAPTER 4

CONCLUSION

A preprocessing method and a recognition method for the pattern recognition machine have been presented. It can be concluded that the pattern recognition machine under consideration has the following distinct advantages.

1) It can recognize the pattern not only under deterministic transformation such as similarity transformation but also under a reasonable amount of random transformation such as change of proportions, bending of lines, noises, etc.

2) The procedure is straightforward, convergent, and mathematically rigorous.

3) The function of the receptor and the categorizer are compatible. Thus, no training is needed.

4) Horizontal and vertical shift of X is equivalent to horizontal and vertical shift of Y_r. The horizontal and vertical shift of Y_r is to produce more than one "proto-type" or typical pattern around which all other patterns in the category cluster. In this sense, the pattern recognition machine under consideration can be viewed as a PWL machine [3].

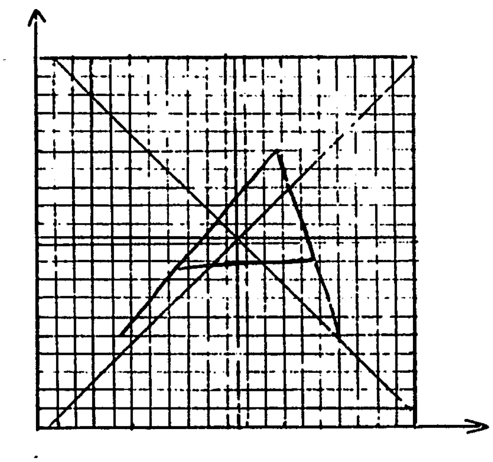


Fig. 1. Sample 'A'.

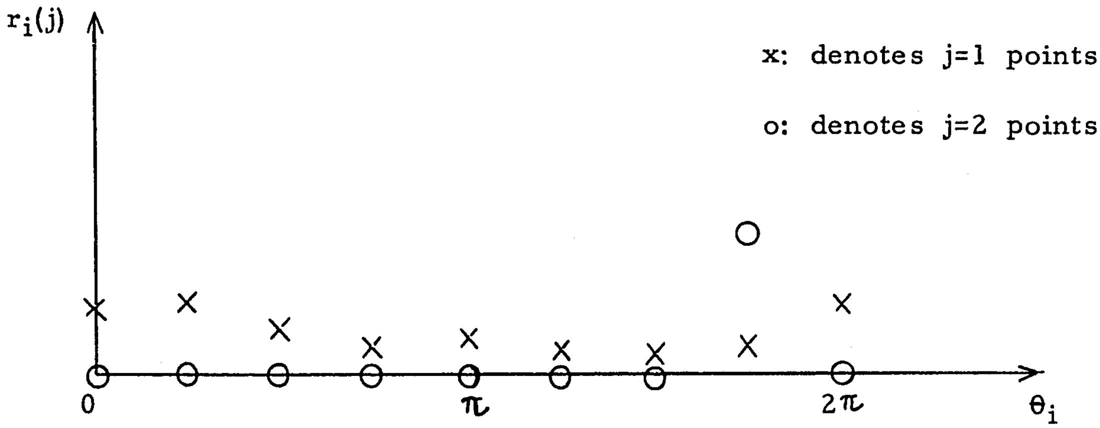


Fig. 2. Quantized transformation curve.

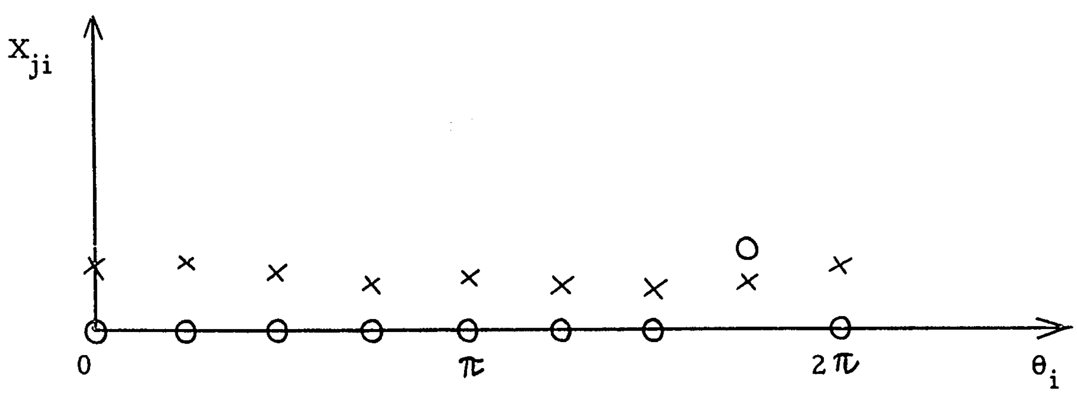


Fig. 3. Quantized logarithmic transformation curve.

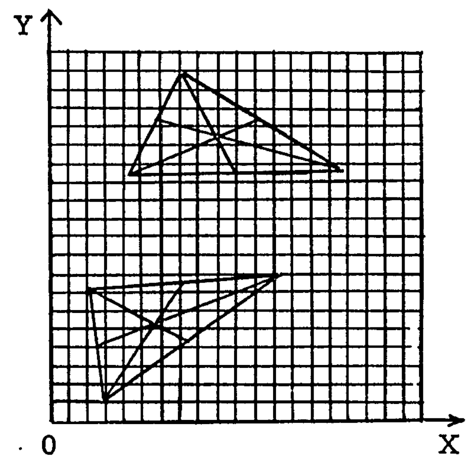


Fig. 4. Pattern under translation and rotation.

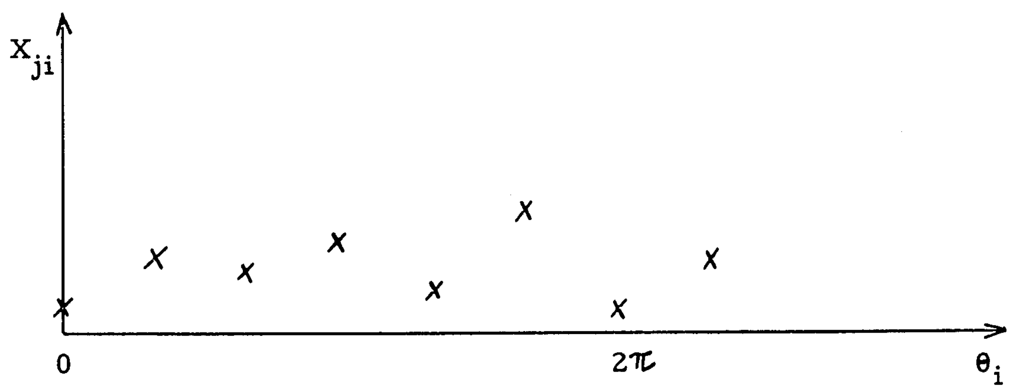


Fig. 5. Quantized curve of the upper triangle.

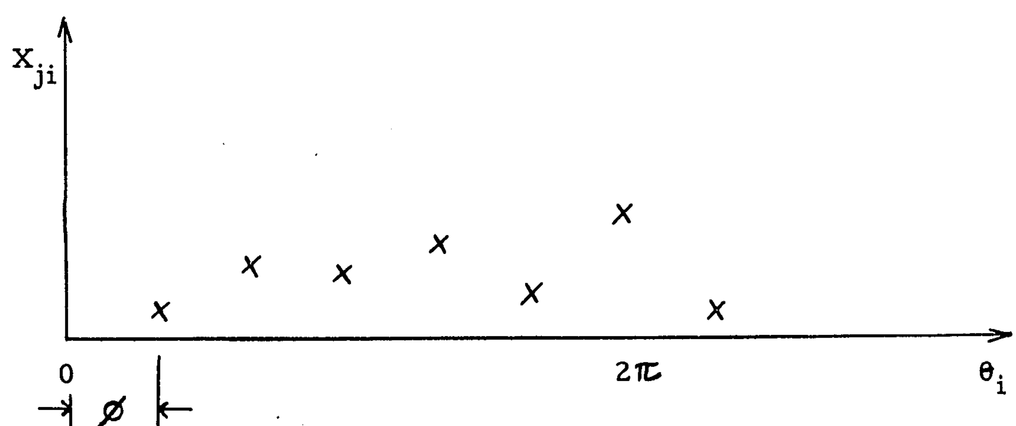


Fig. 6. Quantized curve of the lower triangle.

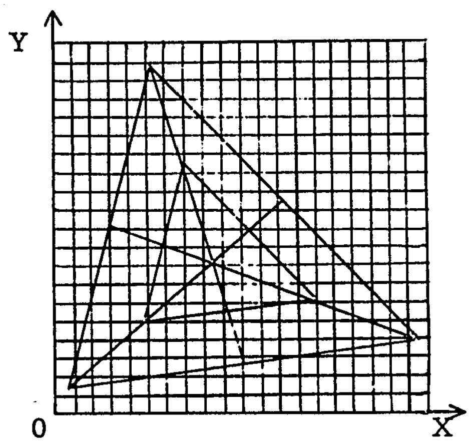


Fig. 7. Pattern under size variation.

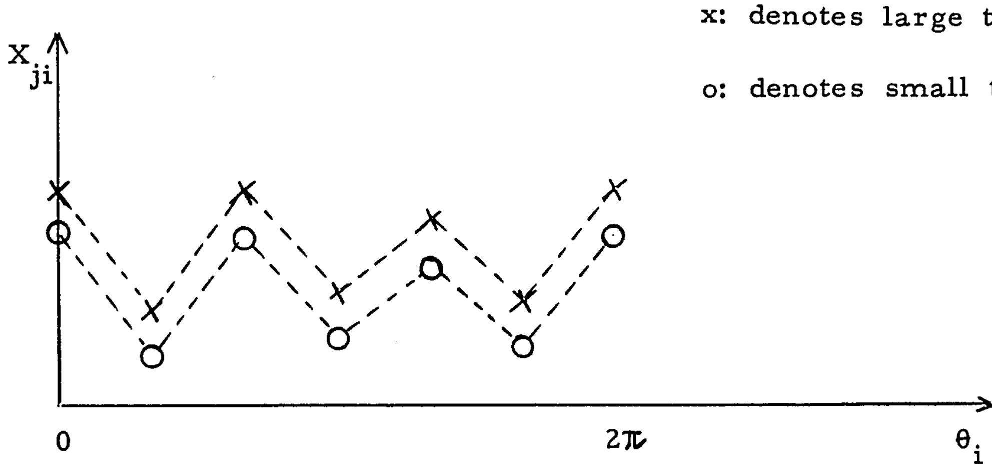


Fig. 8. Quantized curves of two triangles.

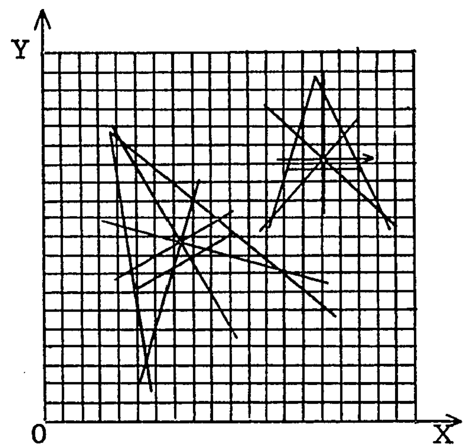


Fig. 9. Pattern class 'A'.

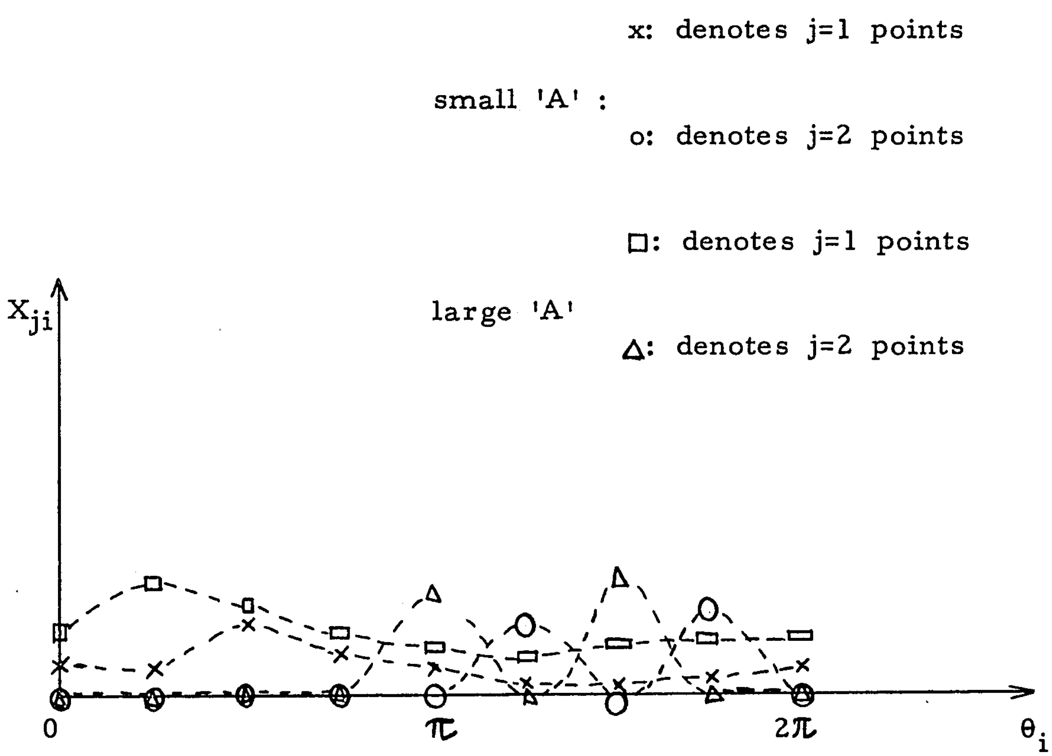


Fig. 10. Multivalued quantized curves.

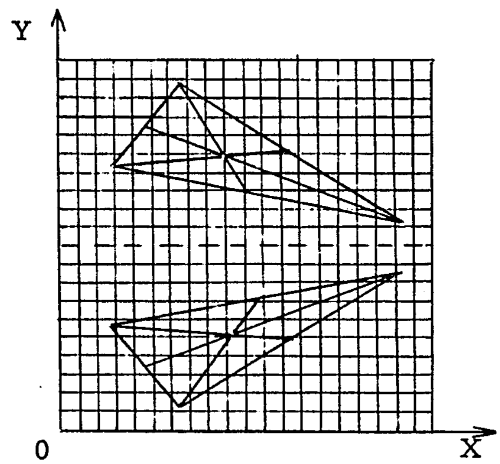


Fig. 11. Image patterns.

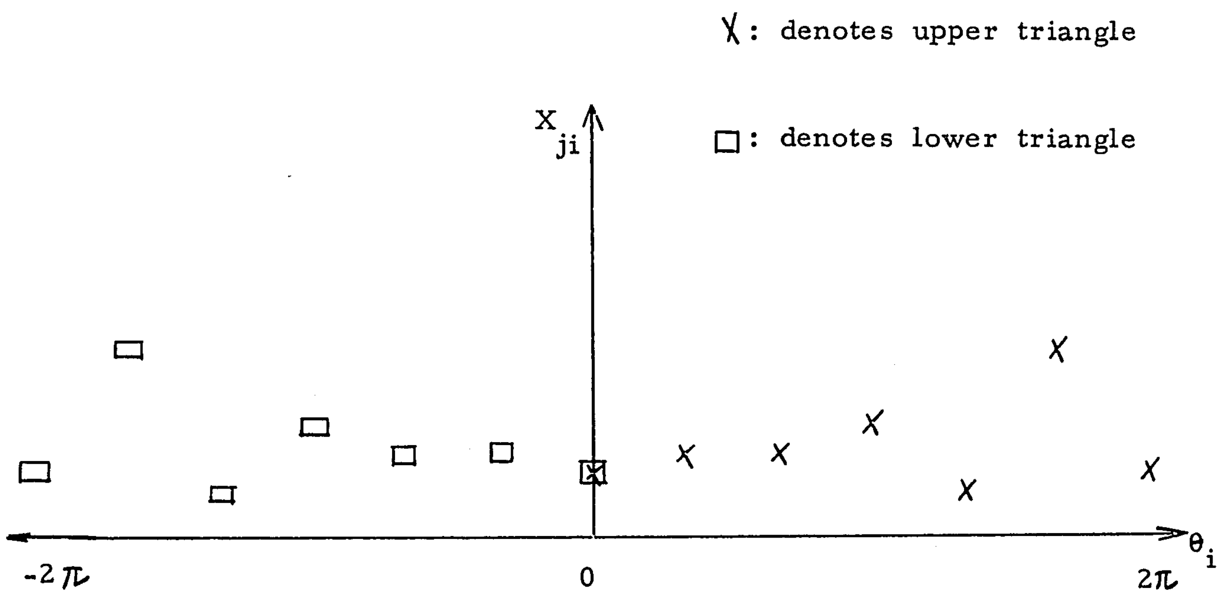


Fig. 12. Quantized curves of image patterns.

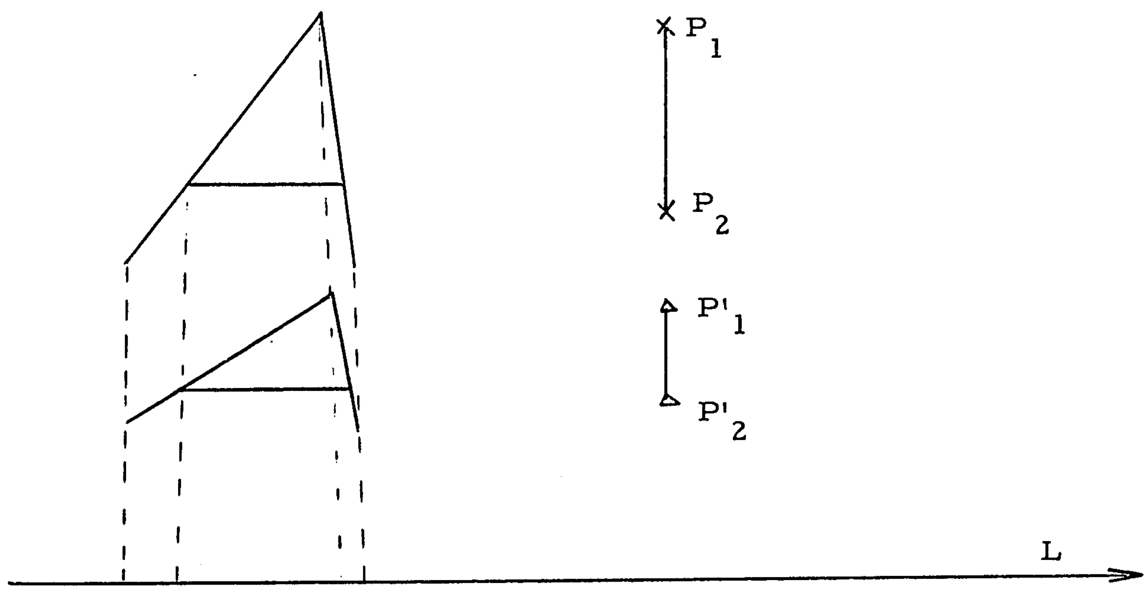


Fig. 13. Squashing of patterns.

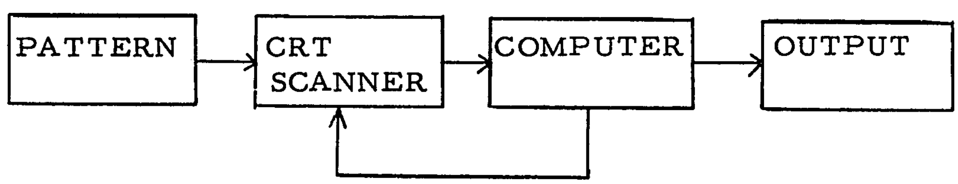


Fig. 14. Proposed receptor.

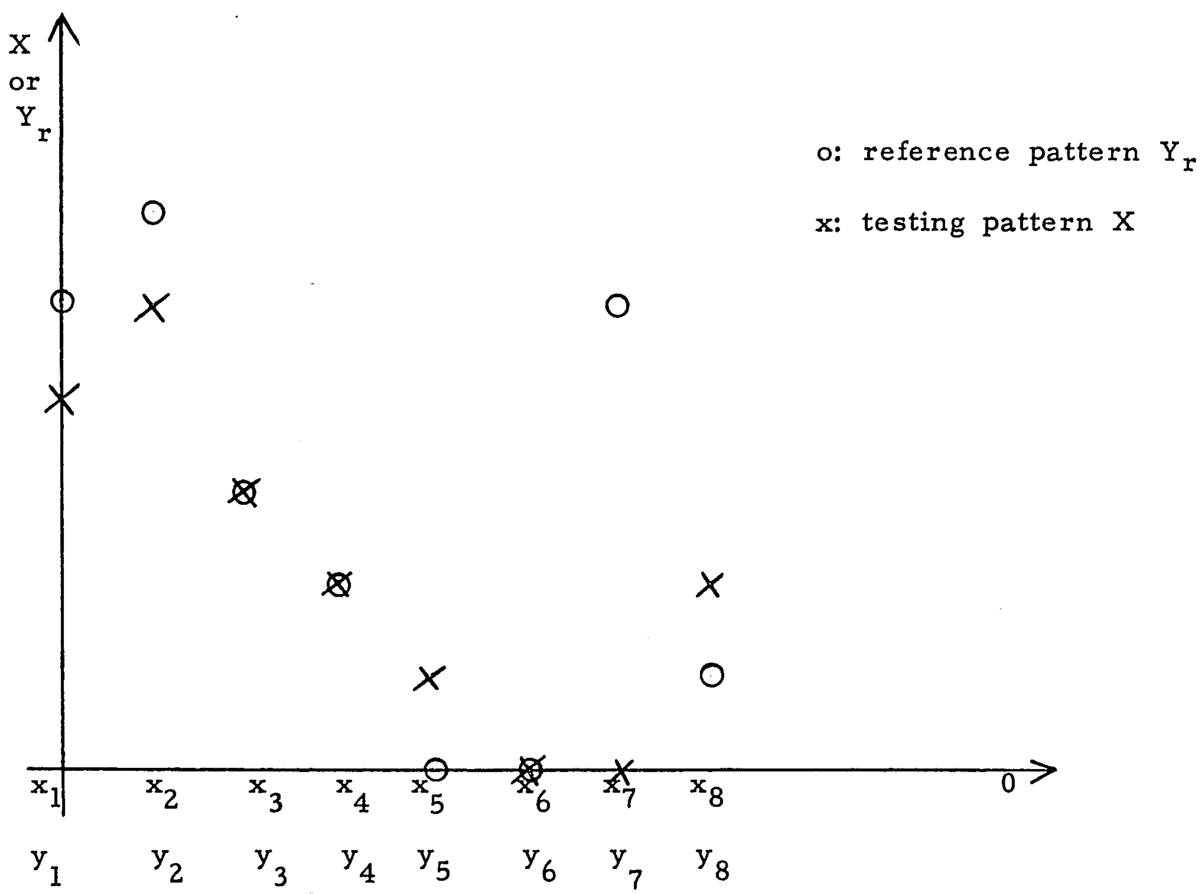
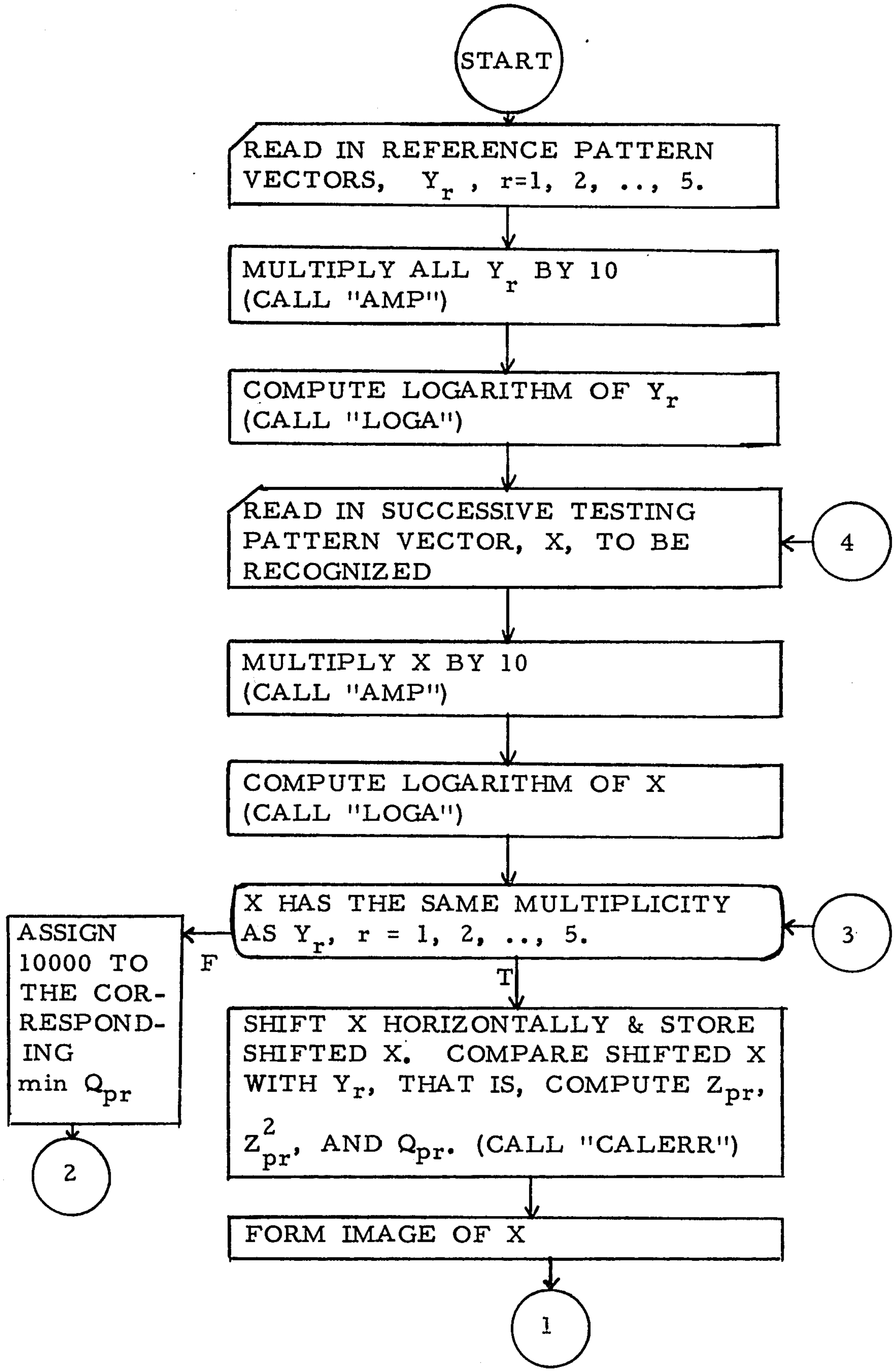


Fig. 15. Relation between Y_r and X, where

$$S_{pr1} = z_1 + z_2 + z_3 + z_4 + z_5 + z_8,$$

$$S_{pr2} = z_7,$$

$$S_{pr3} = z_6.$$


START
READ IN REFERENCE PATTERN VECTORS, Y_r , r=1, 2, .., 5.
MULTIPLY ALL Y_r BY 10 (CALL "AMP")
COMPUTE LOGARITHM OF Y_r (CALL "LOGA")
READ IN SUCCESSIVE TESTING PATTERN VECTOR, X, TO BE RECOGNIZED
4
MULTIPLY X BY 10 (CALL "AMP")
COMPUTE LOGARITHM OF X (CALL "LOGA")
X HAS THE SAME MULTIPLICITY AS Y_r, r = 1, 2, .., 5.
3
F
ASSIGN 10000 TO THE CORRESPONDING min Q_{pr}
2
T
SHIFT X HORIZONTALLY & STORE SHIFTED X. COMPARE SHIFTED X WITH Y_r, THAT IS, COMPUTE Z_{pr}, Z^2_{pr}, AND Q_{pr}. (CALL "CALERR")
FORM IMAGE OF X
1

Continued

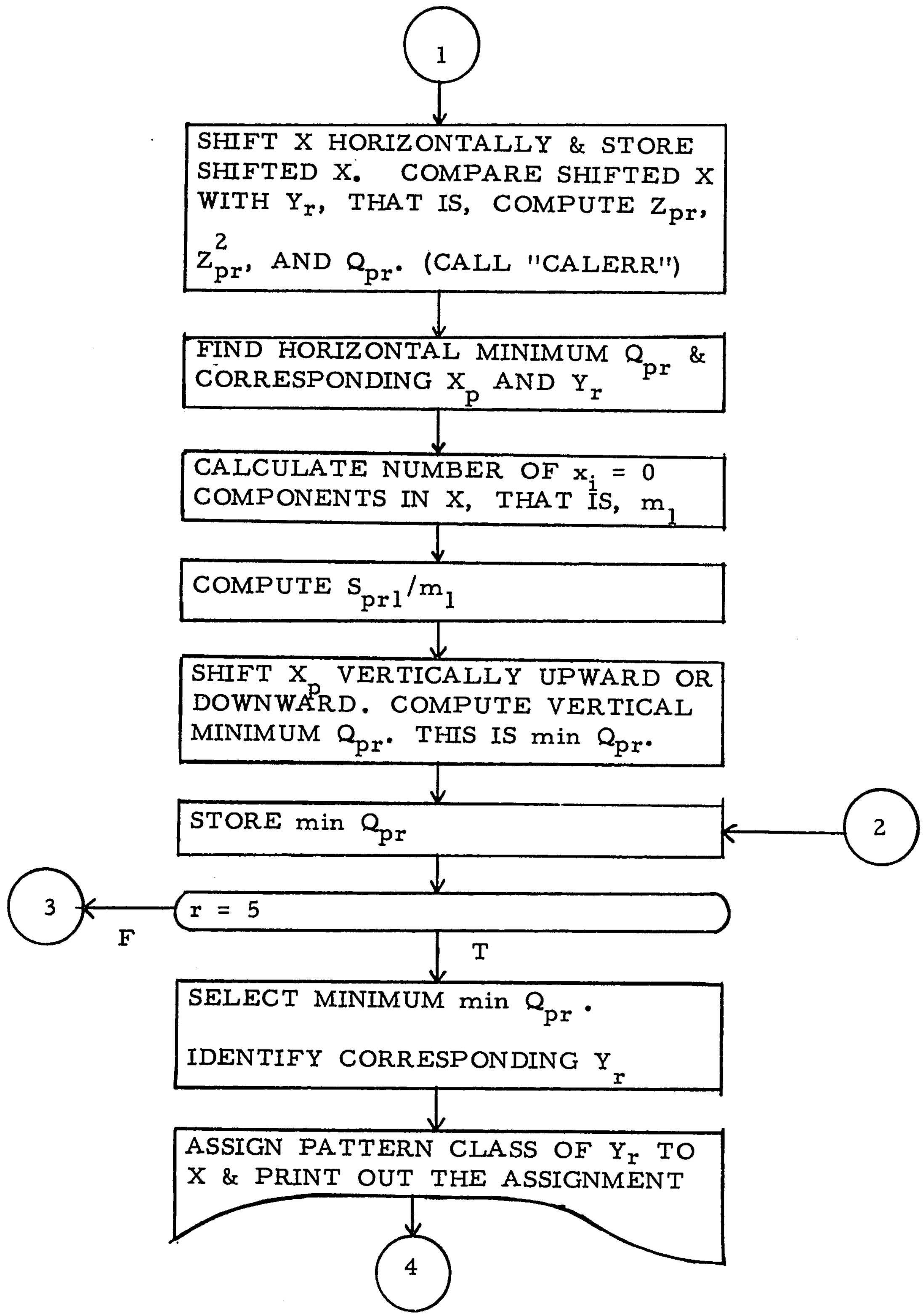


Fig. 16. Simplified flow chart of the main program.

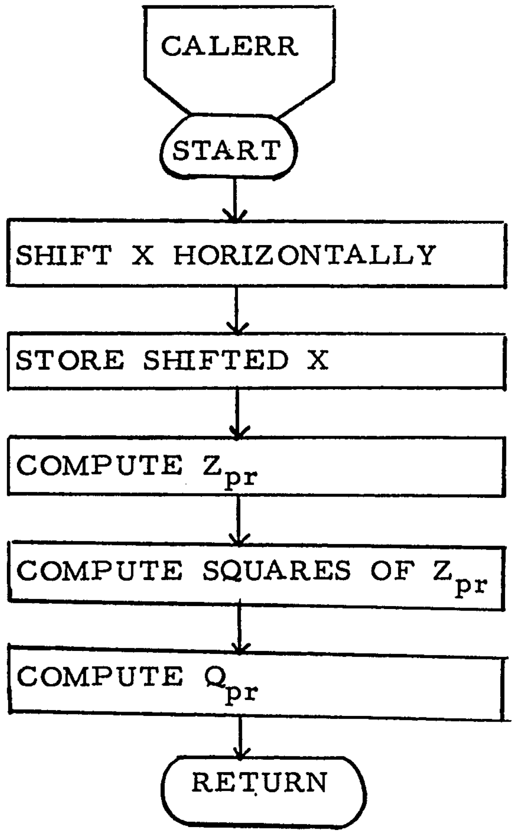
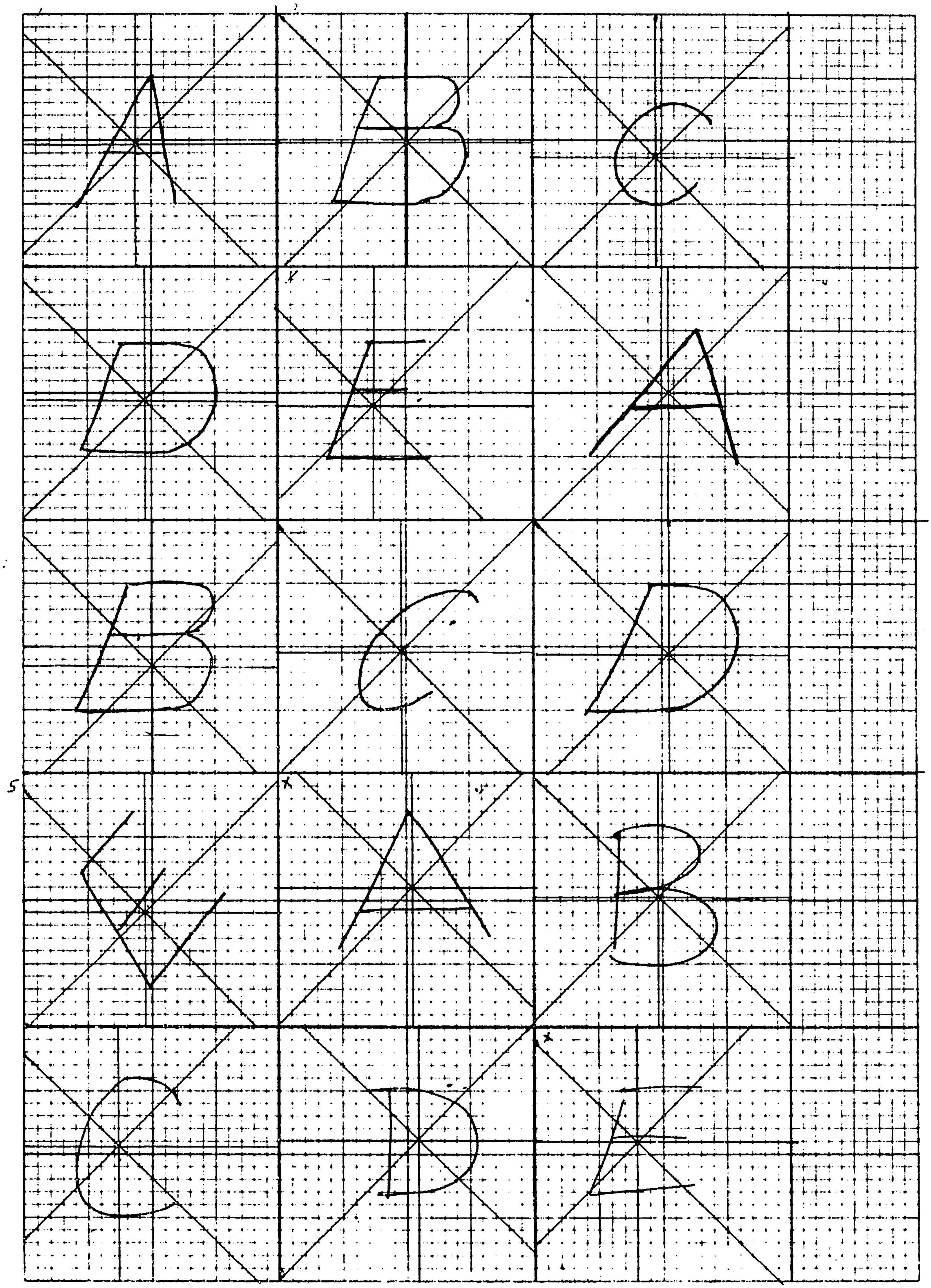
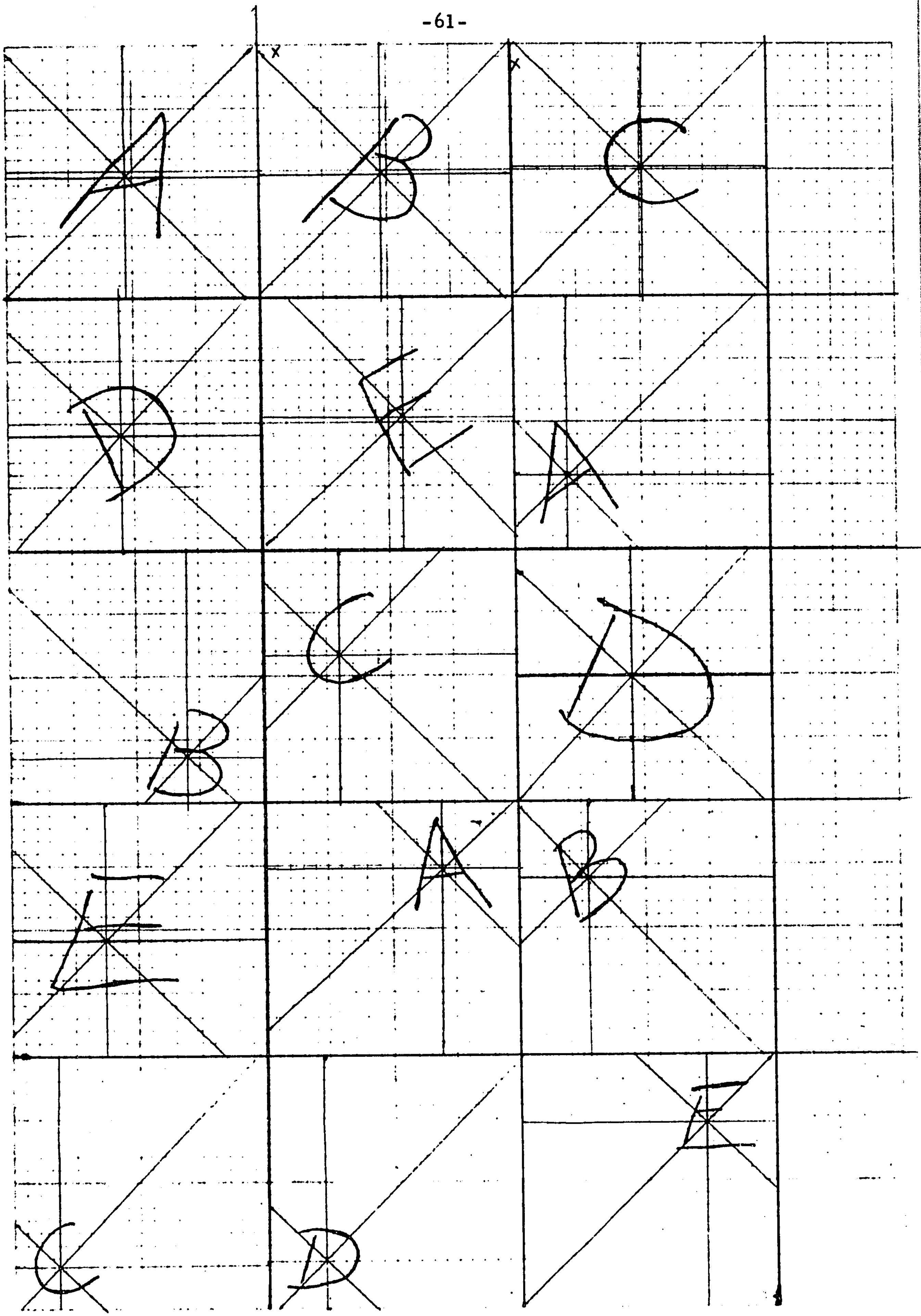


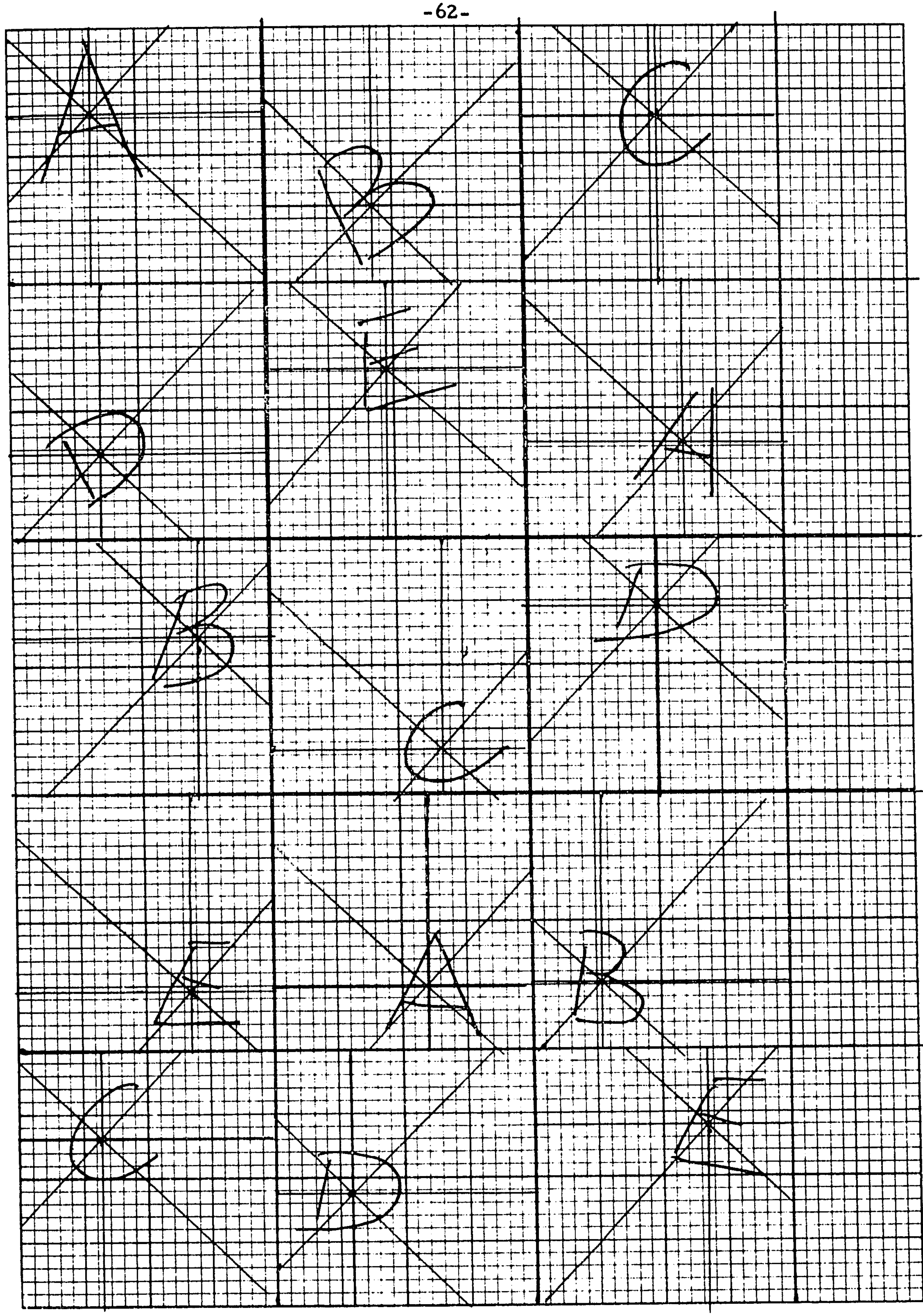
Fig. 17. Flow chart of the subprogram, "CALERR".

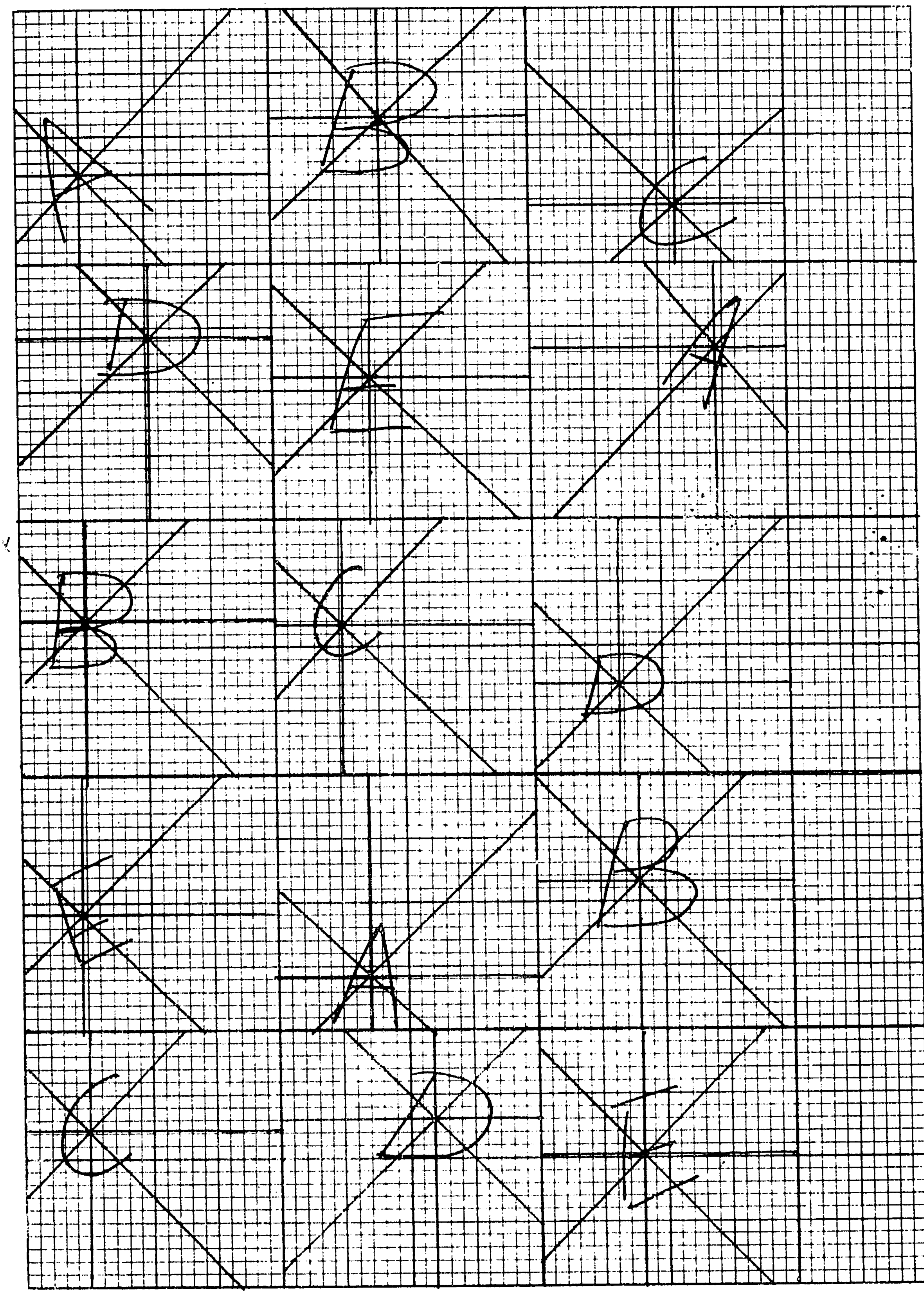
Appendix 1

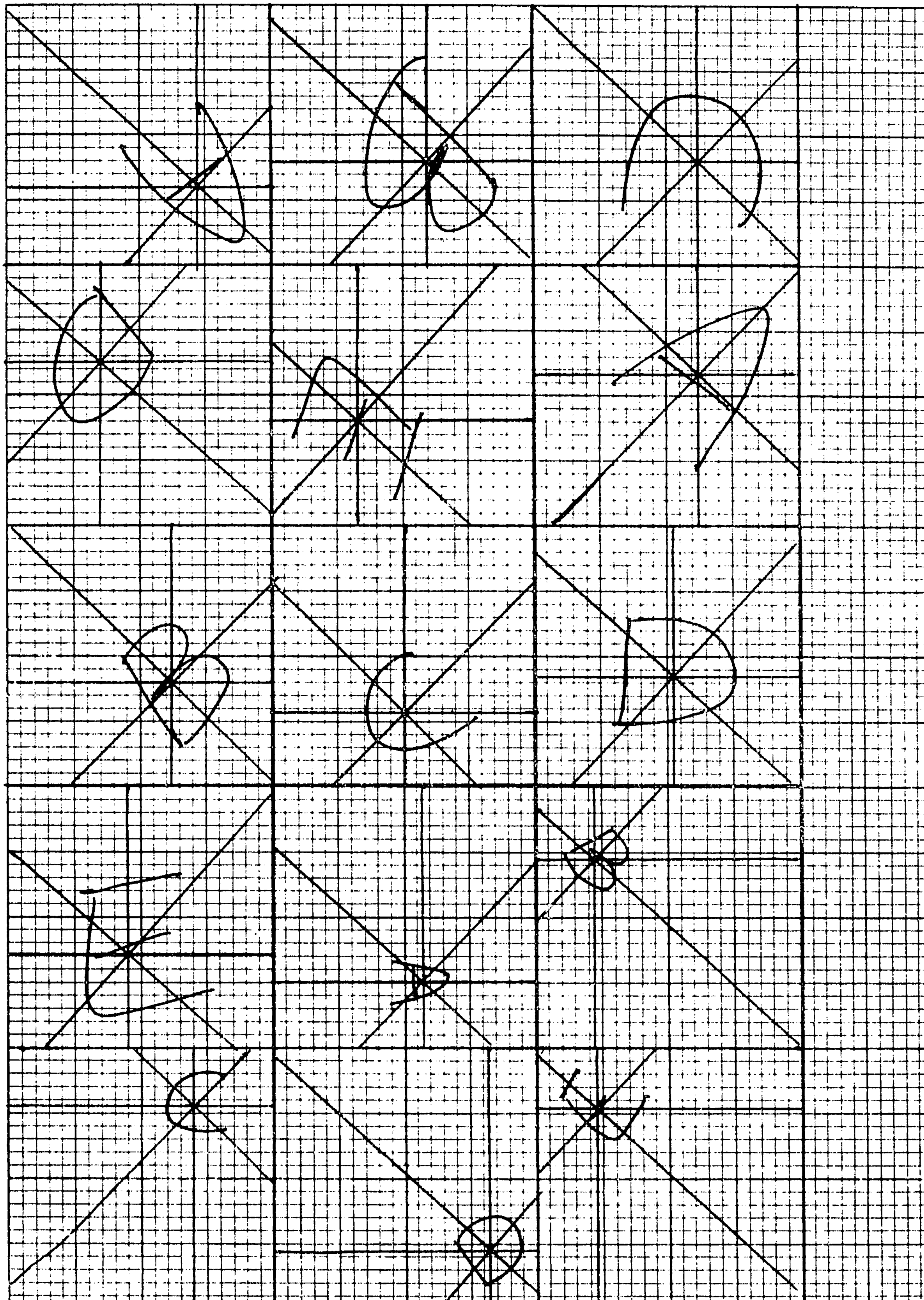




3







Appendix 2


```
               C RECOGNITION PROGRAM
               C PROGRAM FOR THE ALGORITHM OF THE PATTERN RECOGNITION MACHINE
               C PATTERN CLASSES A,B,C,D,E, ARE DESIGNATED AS 1,2,3,4,5
 0001                DIMENSION W(8,2,16), X(8,2,5), Y(8,2,5), Z(8,2,16), SQZ(8,2,16),
                    1SUMSQZ(16), SMALL(5), V(8,2,1)
               C Y STORE FIVE REFERENCE PATTERNS
               C X STORE FIVE TESTING PATTERNS
               C W STORE SIXTEEN ROTATED TESTING PATTERNS
               C Z STORE SIXTEEN ERROR VECTORS
               C SQZ STORE SIXTEEN SQUARES OF THE ERROR VECTORS
               C SUMSQZ STORE SIXTEEN SUM OF THE SQUARES OF THE ERRORS
               C SMALL STORE FIVE MINIMUM SUM OF THE SQUARES OF THE ERRORS
               C SUMZ STORE SUM OF ERRORS
               C N DENOTE REFERENCE PATTERN CLASS
               C K DENOTE TESTING PATTERN CLASS
 0002                READ(1,1) (((Y(L,M,N),L=1,8),M=1,2),N=1,5)
 0003              1 FORMAT (8F10.4)
               C INPUT DATA ARE AMPLIFIED TO AVOID THE CASE OF LOG(1)
 0004                WRITE(3,34)Y
 0005             34 FORMAT(1H1,20X,15H REFERENCE DATA,/,(25X,8F10.4))
 0006                CALL AMP(Y)
 0007                CALL LOGA (Y)
 0008                DO 2 KKK=1,20
 0009                READ(1,1) (((X(I,J,K),I=1,8),J=1,2),K=1,5)
 0010                WRITE(3,35)X
 0011             35 FORMAT(1H1,20X,13H TESTING DATA,/,(25X,8F10.4))
 0012                CALL AMP(X)
 0013                CALL LOGA(X)
 0014                DO 2 K=1,5
 0015                DO 7 N=1,5
               C CLASSIFIED BY THE MULTIPLICITY OF THE PATTERN
 0016                DO 4 L=1,8
 0017                IF (Y(L,2,N).EQ.(0.)) GO TO 4
 0018                GO TO 5
 0019              4 CONTINUE
 0020                DO 6 I=1,8
 0021                IF(X(I,2,K).EQ.(0.)) GO TO 6
 0022                GO TO 3
 0023              6 CONTINUE
               C SHIFTING X AND STORE SHIFTED X IN W
               C CALCULATE Z,SQZ,SUMSQZ
 0024              8 DO 13 KK=1,8
 0025             13 CALL CALERR(W,X,Y,Z,SQZ,SUMSQZ,KK,K,N)
```

```
      C FORMING IMAGE OF THE INPUT PATTERN X
0026        DO 50 J=1,2
0027        DO 50 I=1,8
0028        L=9-I
0029     50 V(L,J,1)=X(I,J,K)
0030        DO 51 J=1,2
0031        DO 51 I=1,8
0032     51 X(I,J,K)=V(I,J,1)
      C SHIFTING X AND STORE SHIFTED IN W
      C CALCULATE Z,SQZ,SUMSQZ
0033        DO 54 KK=9,16
0034     54 CALL CALERR(W,X,Y,Z,SQZ,SUMSQZ,KK,K,N)
      C SELECTING HORIZONTAL MINIMUM SUMSQZ
0035        SMAL=SUMSQZ(1)
0036        DO 16 KK=2,16
0037        IF(SMAL.GT.SUMSQZ(KK))SMAL=SUMSQZ(KK)
0038     16 CONTINUE
      C IDENTIFYING HORIZONTAL MINIMUM SUMSQZ
0039        DO 17 KK=1,16
0040        IF(SMAL.NE.SUMSQZ(KK)) GO TO 17
      C CALCULATING X.NE.0 POINTS
0041        ERRNO=0.
0042        DO 31 J=1,2
0043        DO 31 I=1,8
0044        IF(W(I,J,KK).EQ.(0.))GO TO31
0045        ERRNO=ERRNO+1.
0046     31 CONTINUE
      C FINDING EFFECTIVE SUMZ WHICH CAN BE REDUCED BY VERTICAL SHIFT
0047        SUMZ=0.
0048        DO 18 J=1,2
0049        DO 18 I=1,8
0050        IF(Z(I,J,KK).EQ.(0.))GO TO 18
005'        IF(W(I,J,KK).NE.(0.))SUMZ=SUMZ+Z(I,J,KK)
0052     18 CONTINUE
0053        SUMZ=SUMZ/ERRNO
      C VERTICAL SHIFT UPWARD OR DOWNWARD
0054        IF(SUMZ-0.)21,20,21
0055     21 DO 23 J=1,2
0056        DO 23 I=1,8
0057        IF(W(I,J,KK).EQ.(0.))GO TO 23
0058        W(I,J,KK)=W(I,J,KK)-SUMZ
0059     23 CONTINUE
      C CALCULATING HORIZONTAL AND VERTICAL MINIMUM OF SUMSQZ AS COMPARED WITH
```

```
      C ONE REFERENCE PATTERN CLASS
            DO 25 J=1,2
            DO 25 I=1,8
         25 Z(I,J,KK)=W(I,J,KK)-Y(I,J,N)
            DO 26 J=1,2
            DO 26 I=1,8
         26 SQZ(I,J,KK)=Z(I,J,KK)*Z(I,J,KK)
            SUMSQZ(KK)=0.
            DO 27 J=1,2
            DO 27 I=1,8
         27 SUMSQZ(KK)=SUMSQZ(KK)+SQZ(I,J,KK)
      C HORIZONTAL AND VERTICAL MINIMUM OF SUMSQZ IS STORED IN SMALL
         20 SMALL(N)=SUMSQZ(KK)
         17 CONTINUE
      C COMPARED WITH ANOTHER REFERENCE PATTERN CLASS
            GO TO 7
      C CLASSIFIED BY THE MULTIPLICITY OF THE PATTERN
          5 DO 9 I=1,8
            IF(X(I,2,K).EQ.(0.)) GO TO 9
            GO TO 8
          9 CONTINUE
          3 SMALL(N)=10000.
          7 CONTINUE
      C SELECTING MINIMUM SMALL
            SMIN=SMALL(1)
            DO 28 N=2,5
            IF(SMIN.GT.SMALL(N)) SMIN=SMALL(N)
         28 CONTINUE
            WRITE(3,33)SMIN,SMALL
         33 FORMAT(1X,/,20X,5H SMIN,F10.4,/,20X,6H SMALL,5(1X,F10.4))
      C IDENTIFYING MINIMUM SMALL
            DO 2 N=1,5
            IF(SMIN.NE.SMALL(N)) GO TO 2
      C WRITE TESTING PATTERN CLASS AND IDENTIFIED PATTERN CLASS
            WRITE(3,30) K,N
         30 FORMAT(1X,/,20X,I2,21H INPUT PATTERN CLASS,
           11X,I2,21H OUTPUT PATTERN CLASS)
          2 CONTINUE
      C INPUT ANOTHER SET OF TESTING PATTERNS
            RETURN
            END
```

```
      SUBROUTINE AMP(O)
      DIMENSION O(8,2,5)
      DO 1 K=1,5
      DO 1 J=1,2
      DO 1 I=1,8
    1 O(I,J,K)=10.*O(I,J,K)
      RETURN
      END
```

```
      SUBROUTINE LOGA(A)
      DIMENSION A(8,2,5)
      DO 1 K=1,5
      DO 1 J=1,2
      DO 1 I=1,8
      IF(A(I,J,K).GT.(0.)) GO TO 2
      A(I,J,K)=A(I,J,K)
      GO TO 1
    2 A(I,J,K)=ALOG(A(I,J,K))
    1 CONTINUE
      RETURN
      END
```

```
      SUBROUTINE CALERR(B,C,D,E,F,G,KK,K,N)
      DIMENSION B(8,2,16),C(8,2,5),D(8,2,5),E(8,2,16),F(8,2,16),G(16)
      DO 1 J=1,2
      DO 1 I=1,7
      L=I+1
    1 B(L,J,KK)=C(I,J,K)
      B(1,1,KK)=C(8,1,K)
      B(1,2,KK)=C(8,2,K)
      DO 2 J=1,2
      DO 2 I=1,8
    2 C(I,J,K)=B(I,J,KK)
      DO 3 J=1,2
      DO 3 I=1,8
    3 E(I,J,KK)=C(I,J,K)-D(I,J,N)
      DO 4 J=1,2
      DO 4 I=1,8
    4 F(I,J,KK)=E(I,J,KK)*E(I,J,KK)
      G(KK)=0.
      DO 5 J=1,2
      DO 5 I=1,8
    5 G(KK)=G(KK)+F(I,J,KK)
      RETURN
      END
```

REFERENCE DATA

2.2000	2.6000	3.6000	1.8000	[illegible]	0.8000	0.7000	1.0000
0.0	0.0	[illegible]	[illegible]	[illegible]	[illegible]	0.0	3.3000
4.5000	1.6000	[illegible]	1.7000	[illegible]	[illegible]	4.9000	4.8000
0.0	5.7000	[illegible]	4.5000	[illegible]	[illegible]	0.0	0.0
0.0	6.9000	3.4000	[illegible]	[illegible]	4.6000	4.2000	0.0
0.0	0.0	0.0	[illegible]	0.0	[illegible]	0.0	0.0
5.7000	6.0000	4.8000	[illegible]	3.5000	5.4000	4.0000	4.6000
0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
5.0000	0.0	1.5000	1.0000	1.0000	2.6000	5.3000	3.9000
0.0	0.0	0.0	6.0000	3.1000	0.0	0.0	0.0

TESTING DATA

3.8000	4.0000	2.6000	1.6000	2.0000	1.4000	1.0000	1.5000
0.0	0.0	0.0	0.0	0.0	0.0	0.0	7.4000
4.7000	3.7000	2.8000	3.7000	4.3000	5.0000	3.3000	4.0000
0.0	6.9000	6.9000	4.8000	0.0	0.0	0.0	0.0
0.0	5.0000	4.0000	3.0000	3.0000	3.4000	3.6000	3.6000
0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
5.1000	6.1000	5.7000	3.9000	3.8000	6.2000	4.1000	4.1000
0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
0.0	2.0000	1.3000	1.9000	2.0000	4.4000	4.0000	5.9000
0.0	0.0	5.0000	2.3000	0.0	0.0	0.0	0.0

SMIN 14.8824
SMALL 14.8824 31.8946 10000.0000 10000.0000 22.7504

1 INPUT PATTERN CLASS, 1 OUTPUT PATTERN CLASS

SMIN 2.0776
SMALL 13.7186 2.0776 10000.0000 10000.0000 20.7360

2 INPUT PATTERN CLASS, 2 OUTPUT PATTERN CLASS

SMIN 11.6524
SMALL 10000.0000 10000.0000 11.6524 13.0337 10000.0000

3 INPUT PATTERN CLASS, 3 OUTPUT PATTERN CLASS

SMIN 0.0751
SMALL 10000.0000 10000.0000 17.1976 0.0751 10000.0000

4 INPUT PATTERN CLASS, 4 OUTPUT PATTERN CLASS

SMIN 0.7946
SMALL 26.6590 29.7916 10000.0000 10000.0000 0.7946

5 INPUT PATTERN CLASS, 5 OUTPUT PATTERN CLASS

TESTING DATA

```
        3.8000    3.2000    5.8000    3.2000    3.3000    2.3000    1.8000    2.3000
        0.0       0.0       0.0       0.0       0.0       0.0       0.0       0.0
        3.0000    1.5000    0.8000    1.0000    3.1000    4.8000    5.2000    5.4000
        0.0       4.6000    5.8000    4.5000    0.0       0.0       0.0       0.0
        0.0       6.0000    5.0000    3.3000    3.0000    4.5000    5.4000    0.0
        0.0       0.0       0.0       0.0       0.0       0.0       0.0       0.0
        4.6000    4.1000    4.0000    3.5000    2.2000    3.2000    4.0000    4.4000
        0.0       0.0       0.0       0.0       0.0       0.0       0.0       0.0
        0.0       0.9000    0.5000    0.6000    2.1000    5.1000    3.8000    4.8000
        0.0       6.2000    4.3000    2.4000    0.0       0.0       0.0       0.0

SMIN    0.5833
SMALL 10000.0000 10000.0000    16.1407     0.5833 10000.0000

1 INPUT PATTERN CLASS,  4 OUTPUT PATTERN CLASS

SMIN    0.3949
SMALL    13.5478     0.3949 10000.0000 10000.0000    21.0335

2 INPUT PATTERN CLASS,  2 OUTPUT PATTERN CLASS

SMIN    0.2203
SMALL 10000.0000 10000.0000     0.2203    26.5297 10000.0000

3. INPUT PATTERN CLASS,  3 OUTPUT PATTERN CLASS

SMIN    0.1717
SMALL 10000.0000 10000.0000    16.4979     0.1717 10000.0000

4 INPUT PATTERN CLASS,  4 OUTPUT PATTERN CLASS

SMIN   15.7053
SMALL    23.0169    15.7053 10000.0000 10000.0000    18.0351

5 INPUT PATTERN CLASS,  2 OUTPUT PATTERN CLASS
```

TESTING DATA

3.1000	4.8000	2.2000	1.6000	1.9000	1.3000	0.7000	1.0000
0.0	0.0	0.0	0.0	0.0	0.0	0.0	3.9000
2.8000	1.7000	1.5000	1.8000	2.5000	3.7000	3.2000	3.7000
0.0	5.3000	2.5000	0.0	0.0	0.0	0.0	0.0
0.0	4.5000	3.8000	3.0000	2.7000	2.4000	2.5000	3.8000
0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
4.4000	4.2000	4.0000	4.0000	2.0000	1.7000	3.9000	3.8000
0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
0.0	2.1000	4.8000	0.7000	1.2000	2.0000	3.9000	3.8000
0.0	0.0	0.0	4.2000	2.2000	0.0	0.0	0.0

```
SMIN    15.2198
SMALL     15.2198     31.2559 10000.0000 10000.0000     23.2808

1 INPUT PATTERN CLASS,  1 OUTPUT PATTERN CLASS

SMIN    11.1259
SMALL     21.4788     15.4920 10000.0000 10000.0000     11.1259

2 INPUT PATTERN CLASS,  5 OUTPUT PATTERN CLASS

SMIN    13.2338
SMALL 10000.0000 10000.0000    13.4163    13.2338 10000.0000

3 INPUT PATTERN CLASS,  4 OUTPUT PATTERN CLASS

SMIN     0.4547
SMALL 10000.0000 10000.0000    13.1147     0.4547 10000.0000

4 INPUT PATTERN CLASS,  4 OUTPUT PATTERN CLASS

SMIN    25.6614
SMALL     33.9078     31.0704 10000.0000 10000.0000     25.6614

5 INPUT PATTERN CLASS,  5 OUTPUT PATTERN CLASS
```

TESTING DATA

1.1000	1.8000	2.9000	2.0000	1.5000	2.4000	0.5000	0.5000
2.2000	0.0	0.0	0.0	0.0	0.0	0.0	0.0
2.0000	0.9000	0.8000	2.0000	2.0000	3.5000	2.8000	3.1000
0.0	4.0000	3.8000	1.0000	0.0	0.0	0.0	0.0
0.0	0.0	4.3000	2.8000	2.0000	2.0000	1.9000	2.4000
0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
6.0000	4.7000	5.1000	3.2000	3.3000	5.8000	5.0000	6.0000
0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
0.0	1.8000	1.2000	1.5000	2.7000	5.4000	3.2000	4.8000
0.0	0.0	5.0000	2.8000	0.0	0.0	0.0	0.0

SMIN 15.2101
SMALL 15.2101 30.9201 10000.0000 10000.0000 18.7430

1 INPUT PATTERN CLASS, 1 OUTPUT PATTERN CLASS

SMIN 1.6985
SMALL 15.1066 1.6985 10000.0000 10000.0000 14.1291

2 INPUT PATTERN CLASS, 2 OUTPUT PATTERN CLASS

SMIN 0.3441
SMALL 10000.0000 10000.0000 0.3441 26.5647 10000.0000

3 INPUT PATTERN CLASS, 3 OUTPUT PATTERN CLASS

SMIN 0.2038
SMALL 10000.0000 10000.0000 15.6222 0.2038 10000.0000

4 INPUT PATTERN CLASS, 4 OUTPUT PATTERN CLASS

SMIN 0.3068
SMALL 27.6803 29.1804 10000.0000 10000.0000 0.3068

5 INPUT PATTERN CLASS, 5 OUTPUT PATTERN CLASS

TESTING DATA

1.7000	1.3000	2.9000	1.8000	1.4000	1.0000	0.6000	0.6000
0.0	0.0	0.0	0.0	0.0	2.6000	0.0	0.0
3.0000	2.0000	1.1000	0.9000	1.0000	1.9000	2.9000	2.4000
0.0	0.0	3.7000	3.4000	1.7000	0.0	0.0	0.0
0.0	0.0	3.3000	2.4000	2.0000	2.0000	2.0000	2.2000
0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
2.2000	2.6000	2.5000	1.6000	1.7000	3.0000	1.8000	1.9000
0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
0.0	1.5000	0.9000	1.3000	1.4000	3.0000	2.1000	2.5000
0.0	0.0	2.8000	0.0	0.0	0.0	0.0	0.0

```
SMIN    14.1121
SMALL    14.1121    30.9769 10000.0000 10000.0000    18.6703

1 INPUT PATTERN CLASS,  1 OUTPUT PATTERN CLASS

SMIN    0.5744
SMALL    13.1627     0.5744 10000.0000 10000.0000    17.2815

2 INPUT PATTERN CLASS,  2 OUTPUT PATTERN CLASS

SMIN    0.3268
SMALL 10000.0000 10000.0000     0.3268    26.3396 10000.0000

3 INPUT PATTERN CLASS,  3 OUTPUT PATTERN CLASS

SMIN    0.1120
SMALL 10000.0000 10000.0000    16.7331     0.1120 10000.0000

4 INPUT PATTERN CLASS,  4 OUTPUT PATTERN CLASS

SMIN   12.8183
SMALL    19.9356    39.8512 10000.0000 10000.0000    12.8183

5 INPUT PATTERN CLASS,  5 OUTPUT PATTERN CLASS
```

TESTING DATA

2.0000	2.0000	4.9000	1.8000	1.9000	1.8000	1.1000	1.3000
0.0	0.0	0.0	0.0	0.0	4.3000	0.0	4.2000
4.7000	2.5000	1.3000	1.0000	1.3000	2.9000	4.0000	3.5000
0.0	0.0	4.4000	4.8000	3.1000	0.0	0.0	0.0
0.0	0.0	3.8000	2.7000	2.9000	3.3000	3.9000	4.0000
0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
3.2000	3.8000	3.0000	3.0000	2.3000	2.1000	3.3000	2.9000
0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
0.0	1.8000	0.9000	0.9000	1.6000	2.8000	2.9000	3.2000
0.0	0.0	4.1000	2.2000	0.0	0.0	0.0	0.0

```
SMIN     1.0390
SMALL      1.0390     17.4554 10000.0000 10000.0000     32.3723

1 INPUT PATTERN CLASS,  1 OUTPUT PATTERN CLASS

SMIN     0.3120
SMALL     13.1201      0.3120 10000.0000 10000.0000     19.5617

2 INPUT PATTERN CLASS,  2 OUTPUT PATTERN CLASS

SMIN     0.4041
SMALL 10000.0000 10000.0000     0.4041     26.1379 10000.0000

3 INPUT PATTERN CLASS,  3 OUTPUT PATTERN CLASS

SMIN     0.0929
SMALL 10000.0000 10000.0000    16.8990      0.0929 10000.0000

4 INPUT PATTERN CLASS,  4 OUTPUT PATTERN CLASS

SMIN     0.4862
SMALL     28.8314     29.4769 10000.0000 10000.0000      0.4862

5 INPUT PATTERN CLASS,  5 OUTPUT PATTERN CLASS
```

TESTING DATA

2.2000	3.2000	2.3000	1.3000	1.7000	1.0000	1.0000	1.5000
0.0	0.0	0.0	0.0	0.0	0.0	0.0	3.1000
2.7000	1.3000	1.1000	1.0000	2.1000	5.5000	3.4000	3.0000
0.0	0.0	3.8000	2.3000	0.0	0.0	0.0	0.0
4.5000	0.0	3.3000	2.6000	3.0000	3.0000	2.6000	3.0000
0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
4.4000	4.0000	3.3000	2.4000	2.4000	3.5000	2.2000	2.6000
0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
0.0	1.0000	1.0000	1.6000	1.9000	3.3000	2.6000	3.8000
0.0	0.0	3.5000	0.0	0.0	0.0	0.0	0.0

SMIN 15.2769
SMALL 15.2769 31.6742 10000.0000 10000.0000 21.5759

1 INPUT PATTERN CLASS, 1 OUTPUT PATTERN CLASS

SMIN 11.2383
SMALL 22.6787 15.5004 10000.0000 10000.0000 11.2383

2 INPUT PATTERN CLASS, 5 OUTPUT PATTERN CLASS

SMIN 12.1833
SMALL 10000.0000 10000.0000 12.1833 12.9556 10000.0000

3 INPUT PATTERN CLASS, 3 OUTPUT PATTERN CLASS

SMIN 0.0862
SMALL 10000.0000 10000.0000 17.2016 0.0862 10000.0000

4 INPUT PATTERN CLASS, 4 OUTPUT PATTERN CLASS

SMIN 12.5593
SMALL 19.3219 38.9308 10000.0000 10000.0000 12.5593

5 INPUT PATTERN CLASS, 5 OUTPUT PATTERN CLASS

TESTING DATA

2.0000	2.3000	3.1000	1.4000	1.7000	2.0000	1.6000	2.5000
0.0	0.0	0.0	0.0	0.0	4.5000	0.0	5.0000
1.0000	0.4000	0.6000	1.0000	1.9000	3.0000	3.2000	3.8000
0.0	2.0000	3.8000	2.0000	0.0	0.0	0.0	0.0
0.0	0.0	3.2000	2.2000	2.3000	3.0000	3.1000	3.6000
0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
3.6000	3.6000	3.1000	2.8000	2.4000	3.9000	2.4000	2.9000
0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
2.0000	0.8000	0.7000	1.3000	1.4000	3.6000	3.2000	5.9000
0.0	4.8000	2.4000	0.0	0.0	0.0	0.0	0.0

```
SMIN    2.2755
SMALL     2.2755    18.8802 10000.0000 10000.0000    33.2591

1 INPUT PATTERN CLASS,  1 OUTPUT PATTERN CLASS

SMIN    1.5931
SMALL    16.2841     1.5931 10000.0000 10000.0000    17.0873

2 INPUT PATTERN CLASS,  2 OUTPUT PATTERN CLASS

SMIN    0.4454
SMALL 10000.0000 10000.0000     0.4454    26.1733 10000.0000

3 INPUT PATTERN CLASS,  3 OUTPUT PATTERN CLASS

SMIN    0.0383
SMALL 10000.0000 10000.0000    18.1751     0.0383 10000.0000

4 INPUT PATTERN CLASS,  4 OUTPUT PATTERN CLASS

SMIN    8.3631
SMALL    23.4131    15.7234 10000.0000 10000.0000     8.3631

5 INPUT PATTERN CLASS,  5 OUTPUT PATTERN CLASS
```

```
TESTING DATA
       1.7000     1.8000     2.3000     3.5000     2.3000     1.7000     0.9000     0.9000
       2.7000     0.0        0.0        0.0        0.0        2.8000     0.0        0.0
       2.3000     5.0000     4.3000     3.7000     3.1000     1.0000     0.8000     0.9000
       0.0        0.0        0.0        0.0        0.0        6.0000     4.0000     2.7000
       0.0        0.0        2.8000     2.0000     2.3000     3.2000     3.0000     3.3000
       0.0        0.0        0.0        0.0        0.0        0.0        0.0        0.0
       4.0000     3.4000     4.0000     3.2000     2.5000     4.0000     2.7000     3.0000
       0.0        0.0        0.0        0.0        0.0        0.0        0.0        0.0
       0.0        7.4000     4.0000     2.2000     2.2000     0.9000     0.6000     1.1000
       0.0        0.0        0.0        0.0        0.0        4.2000     4.1000     0.0

SMIN   24.7268
SMALL     24.7268     44.9045 10000.0000 10000.0000     29.8589

1 INPUT PATTERN CLASS,  1 OUTPUT PATTERN CLASS

SMIN    0.4227
SMALL     12.0857      0.4227 10000.0000 10000.0000     18.6460

2 INPUT PATTERN CLASS,  2 OUTPUT PATTERN CLASS

SMIN    0.6502
SMALL 10000.0000 10000.0000      0.6502     26.1414 10000.0000

3 INPUT PATTERN CLASS,  3 OUTPUT PATTERN CLASS

SMIN    0.1084
SMALL 100CC.0000 10000.0000     18.3148      0.1084 10000.0000

4 INPUT PATTERN CLASS,  4 OUTPUT PATTERN CLASS

SMIN    1.2647
SMALL     33.5427     29.1395 10000.0000 10000.0000      1.2647

5 INPUT PATTERN CLASS,  5 OUTPUT PATTERN CLASS
```

TESTING DATA

0.6000	1.5000	2.1000	1.4000	2.1000	1.4000	1.2000	1.0000
0.0	0.0	0.0	0.0	0.0	0.0	3.0000	0.0
2.0000	4.1000	3.8000	3.0000	2.3000	0.8000	0.6000	0.7000
0.0	0.0	0.0	0.0	0.0	3.7000	3.5000	3.2000
0.0	0.0	4.0000	2.6000	2.2000	2.5000	2.3000	2.1000
0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
3.3000	3.0000	2.2000	2.1000	2.0000	3.4000	2.2000	2.3000
0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
0.0	1.5000	3.6000	3.4000	1.5000	1.3000	1.2000	1.1000
0.0	0.0	0.0	0.0	0.0	0.0	3.8000	3.5000

SMIN 14.9603
SMALL 14.9603 31.2864 10000.0000 10000.0000 18.9281

1 INPUT PATTERN CLASS, 1 OUTPUT PATTERN CLASS

SMIN 0.3773
SMALL 12.4546 0.3773 10000.0000 10000.0000 21.3421

2 INPUT PATTERN CLASS, 2 OUTPUT PATTERN CLASS

SMIN 0.3714
SMALL 10000.0000 10000.0000 0.3714 26.3920 10000.0000

3 INPUT PATTERN CLASS, 3 OUTPUT PATTERN CLASS

SMIN 0.0706
SMALL 10000.0000 10000.0000 17.6425 0.0706 10000.0000

4 INPUT PATTERN CLASS, 4 OUTPUT PATTERN CLASS

SMIN 17.2359
SMALL 30.9898 24.0313 10000.0000 10000.0000 17.2359

5 INPUT PATTERN CLASS, 5 OUTPUT PATTERN CLASS

```
TESTING DATA
        1.3000     2.0000     3.0000     1.2000     1.5000     1.5000     0.9000     1.3000
        0.0        0.0        0.0        0.0        0.0        3.0000     0.0        2.6000
        4.1000     1.5000     1.1000     1.3000     2.4000     4.5000     3.4000     4.0000
        0.0        4.0000     4.9000     2.8000     0.0        0.0        0.0        0.0
        0.0        0.0        3.5000     2.4000     2.1000     2.9000     3.3000     3.2000
        0.0        0.0        0.0        0.0        0.0        0.0        0.0        0.0
        4.2000     4.0000     3.8000     2.1000     2.3000     4.0000     3.0000     3.2000
        0.0        0.0        0.0        0.0        0.0        0.0        0.0        0.0
        0.0        0.2000     0.2000     0.2000     1.8000     2.3000     3.8000     3.4000
        0.0        0.0        4.6000     2.3000     0.0        0.0        0.0        0.0

SMIN    1.5127
SMALL     1.5127    17.4511 10000.0000 10000.0000    31.9709

1 INPUT PATTERN CLASS,  1 OUTPUT PATTERN CLASS

SMIN    0.3548
SMALL    13.5687     0.3548 10000.0000 10000.0000    17.5522

2 INPUT PATTERN CLASS,  2 OUTPUT PATTERN CLASS

SMIN    0.2651
SMALL 10000.0000 10000.0000     0.2651    26.1691 10000.0000

3 INPUT PATTERN CLASS,  3 OUTPUT PATTERN CLASS

SMIN    0.1002
SMALL 10000.0000 10000.0000    15.2704     0.1002 10000.0000

4 INPUT PATTERN CLASS,  4 OUTPUT PATTERN CLASS

SMIN    4.2772
SMALL    37.2257    32.9655 10000.0000 10000.0000     4.2772

5 INPUT PATTERN CLASS,  5 OUTPUT PATTERN CLASS
```

TESTING DATA

2.8000	5.5000	3.5000	2.9000	2.7000	1.8000	2.5000	3.0000
0.0	0.0	0.0	0.0	4.0000	0.0	7.5000	0.0
0.2000	0.7000	0.4000	0.5000	3.7000	4.5000	4.0000	4.7000
0.0	4.9000	4.3000	3.0000	0.0	0.0	0.0	0.0
0.0	0.0	3.2000	2.9000	3.1000	3.3000	3.1000	4.7000
0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
4.8000	5.5000	5.0000	3.3000	4.2000	4.3000	4.8000	4.0000
0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
0.0	1.9000	0.9000	1.0000	2.9000	3.7000	3.0000	3.1000
0.0	0.0	4.6000	3.0000	0.0	0.0	0.0	0.0

SMIN 1.2238
SMALL 1.2238 17.1860 10000.0000 10000.0000 30.7490

1 INPUT PATTERN CLASS, 1 OUTPUT PATTERN CLASS

SMIN 7.7893
SMALL 21.3262 7.7893 10000.0000 10000.0000 17.4317

2 INPUT PATTERN CLASS, 2 OUTPUT PATTERN CLASS

SMIN 0.4363
SMALL 10000.0000 10000.0000 0.4353 26.3060 10000.0000

3 INPUT PATTERN CLASS, 3 OUTPUT PATTERN CLASS

SMIN 0.1858
SMALL 10000.0000 10000.0000 18.1396 0.1858 10000.0000

4 INPUT PATTERN CLASS, 4 OUTPUT PATTERN CLASS

SMIN 0.4281
SMALL 29.6104 29.0438 10000.0000 10000.0000 0.4281

5 INPUT PATTERN CLASS, 5 OUTPUT PATTERN CLASS

```
TESTING DATA
        3.1000    3.0000    5.4000    4.9000    4.2000    2.2000    1.0000    1.2000
        0.0       0.0       0.0       0.0       0.0       5.8000    0.0       7.8000
        4.5000    1.2000    1.1000    0.8000    2.0000    4.8000    4.0000    4.3000
        0.0       0.0       5.0000    5.3000    3.9000    0.0       0.0       0.0
        0.0       0.0       3.9000    2.7000    2.9000    3.2000    3.3000    4.6000
        0.0       0.0       0.0       0.0       0.0       0.0       0.0       0.0
        3.2000    3.6000    2.9000    2.6000    3.2000    2.8000    4.5000    3.3000
        0.0       0.0       0.0       0.0       0.0       0.0       0.0       0.0
        0.9000    0.0       4.7000    4.5000    2.5000    1.8000    0.6000    0.4000
        0.0       0.0       0.0       0.0       0.0       2.8000    4.1000    4.0000

SMIN    0.2230
SMALL     0.2230    16.1951 10000.0000 10000.0000    32.1944

1 INPUT PATTERN CLASS,  1 OUTPUT PATTERN CLASS

SMIN    1.6251
SMALL    14.0642     1.6251 10000.0000 10000.0000    18.2118

2 INPUT PATTERN CLASS,  2 OUTPUT PATTERN CLASS

SMIN    0.4425
SMALL 10000.0000 10000.0000     0.4425    26.2699 10000.0000

3 INPUT PATTERN CLASS,  3 OUTPUT PATTERN CLASS

SMIN    0.2315
SMALL 10000.0000 10000.0000    18.6390     0.2315 10000.0000

4 INPUT PATTERN CLASS,  4 OUTPUT PATTERN CLASS

SMIN   13.4265
SMALL    23.3351    17.6264 10000.0000 10000.0000    13.4265

5 INPUT PATTERN CLASS,  5 OUTPUT PATTERN CLASS
```

TESTING DATA

1.0000	2.6000	4.0000	3.8000	3.3000	1.2000	0.3000	0.4000
0.0	0.0	0.0	0.0	0.0	5.0000	0.0	2.6000
4.7000	4.3000	0.4000	0.4000	0.3000	0.3000	3.0000	4.5000
0.0	0.0	3.0000	5.3000	4.2000	2.4000	0.0	0.0
0.0	0.0	3.8000	2.7000	2.5000	2.7000	3.1000	5.1000
0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
4.5000	5.1000	4.5000	4.0000	4.2000	3.2000	4.7000	3.7000
0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
2.0000	0.0	4.7000	3.0000	1.6000	2.0000	1.3000	1.2000
0.0	0.0	0.0	0.0	0.0	0.0	5.0000	5.6000

SMIN 2.7030
SMALL 2.7030 18.0931 10000.0000 10000.0000 30.8548

1 INPUT PATTERN CLASS, 1 OUTPUT PATTERN CLASS

SMIN 21.7003
SMALL 30.5627 21.7003 10000.0000 10000.0000 28.4761

2 INPUT PATTERN CLASS, 2 OUTPUT PATTERN CLASS

SMIN 0.2431
SMALL 10000.0000 10000.0000 0.2431 26.4940 10000.0000

3 INPUT PATTERN CLASS, 3 OUTPUT PATTERN CLASS

SMIN 0.0805
SMALL 10000.0000 10000.0000 18.7999 0.0805 10000.0000

4 INPUT PATTERN CLASS, 4 OUTPUT PATTERN CLASS

SMIN 1.9482
SMALL 33.6110 29.9522 10000.0000 10000.0000 1.9482

5 INPUT PATTERN CLASS, 5 OUTPUT PATTERN CLASS

TESTING DATA

2.1000	3.5000	2.2000	1.6000	2.4000	0.3000	0.2000	0.6000
0.0	0.0	0.0	0.0	0.0	0.0	0.0	2.4000
1.5000	1.0000	1.2000	1.0000	1.6000	3.0000	2.1000	2.2000
0.0	3.2000	2.5000	1.1000	0.0	0.0	0.0	0.0
0.0	7.5000	3.8000	2.8000	3.0000	3.5000	3.2000	4.4000
0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
4.2000	4.8000	3.9000	2.8000	2.4000	5.0000	3.8000	3.4000
0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
0.0	1.0000	0.5000	0.5000	2.9000	6.0000	2.9000	3.3000
0.0	0.0	5.5000	3.0000	0.0	0.0	0.0	0.0

SMIN 16.5182
SMALL 16.5182 31.9228 10000.0000 10000.0000 19.2925

1 INPUT PATTERN CLASS, 1 OUTPUT PATTERN CLASS

SMIN 1.4683
SMALL 12.9732 1.4683 10000.0000 10000.0000 15.9052

2 INPUT PATTERN CLASS, 2 OUTPUT PATTERN CLASS

SMIN 12.6392
SMALL 10000.0000 10000.0000 12.6392 13.4734 10000.0000

3 INPUT PATTERN CLASS, 3 OUTPUT PATTERN CLASS

SMIN 0.0986
SMALL 10000.0000 10000.0000 15.4841 0.0986 10000.0000

4 INPUT PATTERN CLASS, 4 OUTPUT PATTERN CLASS

SMIN 0.7870
SMALL 31.6441 29.8948 10000.0000 10000.0000 0.7870

5 INPUT PATTERN CLASS, 5 OUTPUT PATTERN CLASS

TESTING DATA

3.5000	2.7000	2.4000	2.0000	1.1000	1.3000	1.5000	1.4000
0.0	0.0	0.0	4.0000	0.0	4.6000	0.0	0.0
3.5000	6.2000	5.3000	4.0000	4.0000	0.8000	0.6000	1.0000
0.0	0.0	0.0	0.0	0.0	6.2000	5.2000	5.9000
0.0	0.0	4.4000	3.4000	2.9000	2.1000	2.1000	3.9000
0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
4.3000	6.3000	5.5000	4.0000	3.4000	7.0000	3.9000	2.2000
0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
0.0	0.8000	0.3000	0.3000	0.7000	3.5000	3.9000	4.0000
0.0	0.0	5.0000	3.1000	2.6000	0.0	0.0	0.0

```
SMIN    1.0174
SMALL     1.0174    17.5025 10000.0000 10000.0000    31.3183

1 INPUT PATTERN CLASS,  1 OUTPUT PATTERN CLASS

SMIN    0.9685
SMALL    12.9843     0.9685 10000.0000 10000.0000    25.2374

2 INPUT PATTERN CLASS,  2 OUTPUT PATTERN CLASS

SMIN    0.3227
SMALL 10000.0000 10000.0000     0.3227    26.3991 10000.0000

3 INPUT PATTERN CLASS,  3 OUTPUT PATTERN CLASS

SMIN    0.4229
SMALL 10000.0000 10000.0000    15.2224     0.4229 10000.0000

4 INPUT PATTERN CLASS,  4 OUTPUT PATTERN CLASS

SMIN   15.5213
SMALL    23.2066    18.2789 10000.0000 10000.0000    15.5213

5 INPUT PATTERN CLASS,  5 OUTPUT PATTERN CLASS
```

TESTING DATA

1.8000	3.0000	3.5000	4.8000	2.4000	2.3000	2.7000	1.4000
5.0000	0.0	0.0	0.0	0.0	0.0	5.0000	0.0
5.0000	6.4000	3.5000	1.2000	1.0000	1.4000	3.5000	5.6000
0.0	0.0	0.0	5.5000	6.1000	3.5000	0.0	0.0
0.0	5.3000	3.9000	3.3000	3.1000	3.3000	4.6000	0.0
0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
4.7000	5.0000	4.8000	3.5000	3.4000	5.0000	4.0000	4.0000
0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
0.0	0.0	0.3000	0.2000	0.5000	2.4000	4.5000	4.0000
0.0	0.0	5.0000	4.5000	2.2000	0.0	0.0	0.0

SMIN 1.2299
SMALL 1.2299 17.0119 10000.0000 10000.0000 31.7133

1 INPUT PATTERN CLASS, 1 OUTPUT PATTERN CLASS

SMIN 0.3440
SMALL 13.6761 0.3440 10000.0000 10000.0000 19.0263

2 INPUT PATTERN CLASS, 2 OUTPUT PATTERN CLASS

SMIN 0.3128
SMALL 10000.0000 10000.0000 0.3128 26.3343 10000.0000

3 INPUT PATTERN CLASS, 3 OUTPUT PATTERN CLASS

SMIN 0.0420
SMALL 10000.0000 10000.0000 17.4571 0.0420 10000.0000

4 INPUT PATTERN CLASS, 4 OUTPUT PATTERN CLASS

SMIN 24.0477
SMALL 35.0896 32.7227 10000.0000 10000.0000 24.0477

5 INPUT PATTERN CLASS, 5 OUTPUT PATTERN CLASS

```
TESTING DATA
        3.1000    3.0000    1.0000    0.8000    1.0000    2.5000    2.9000    4.7000
        0.0       0.0       6.5000    0.0       3.4000    0.0       0.0       0.0
        1.0000    2.0000    4.0000    5.4000    4.6000    4.7000    1.7000    1.0000
        3.8000    2.5000    0.0       0.0       0.0       0.0       0.0       6.0000
        4.8000    4.8000    5.0000    5.1000    5.0000    0.0       0.0       5.5000
        0.0       0.0       0.0       0.0       0.0       0.0       0.0       0.0
        4.0000    3.4000    5.2000    3.7000    3.3000    4.4000    4.1000    3.1000
        0.0       0.0       0.0       0.0       0.0       0.0       0.0       0.0
        0.2000    0.3000    3.5000    4.7000    4.2000    0.0       0.2000    0.2000
        4.0000    2.5000    0.0       0.0       0.0       0.0       0.0       5.0000

SMIN    0.3552
SMALL     0.3552     16.0838 10000.0000 10000.0000     28.7863

1 INPUT PATTERN CLASS,  1 OUTPUT PATTERN CLASS

SMIN    1.1001
SMALL    11.6040      1.1001 10000.0000 10000.0000     20.3469

2 INPUT PATTERN CLASS,  2 OUTPUT PATTERN CLASS

SMIN    0.5439
SMALL 10000.0000 10000.0000      0.5439    26.1428 10000.0000

3 INPUT PATTERN CLASS,  3 OUTPUT PATTERN CLASS

SMIN    0.2305
SMALL 10000.0000 10000.0000     18.9025      0.2305 10000.0000

4 INPUT PATTERN CLASS,  4 OUTPUT PATTERN CLASS

SMIN   22.5715
SMALL    32.4470     24.9870 10000.0000 10000.0000     22.5715

5 INPUT PATTERN CLASS,  5 OUTPUT PATTERN CLASS
```

TESTING DATA

4.2000	7.2000	3.6000	3.4000	1.0000	0.5000	0.8000	3.8000
0.0	0.0	0.C	0.0	5.5000	0.0	7.1000	0.0
4.1000	3.0000	0.8000	0.6000	0.7000	2.2000	4.0000	3.8000
0.0	0.0	4.5000	4.2000	2.6000	0.0	0.0	0.0
0.0	0.0	4.6000	3.4000	3.0000	3.0000	2.9000	3.2000
0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
4.6000	4.6000	4.4000	4.9000	3.7000	5.0000	3.2000	3.5000
0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
0.0	1.4000	0.9000	0.9000	1.5000	4.6000	4.2000	5.0000
0.0	0.0	5.7000	4.0000	3.0000	0.0	0.0	0.0

SMIN 1.2311
SMALL 1.2311 16.9852 10000.0000 10000.0000 32.0592

1 INPUT PATTERN CLASS, 1 OUTPUT PATTERN CLASS

SMIN 0.5588
SMALL 13.7226 0.5588 10000.0000 10000.0000 19.6182

2 INPUT PATTERN CLASS, 2 OUTPUT PATTERN CLASS

SMIN 0.3726
SMALL 10000.0000 10000.0000 0.3726 26.2961 10000.0000

3 INPUT PATTERN CLASS, 3 OUTPUT PATTERN CLASS

SMIN 0.0834
SMALL 10000.0000 10000.0000 17.4439 0.0834 10000.0000

4 INPUT PATTERN CLASS, 4 OUTPUT PATTERN CLASS

SMIN [illegible].576
SMALL 20.3038 15.2334 10000.0000 10000.0000 11.2578

5 INPUT PATTERN CLASS, 5 OUTPUT PATTERN CLASS

```
TESTING DATA
          1.9000    1.5000    1.3000    1.3000    0.9000    1.0000    1.0000    1.0000
          0.0       0.0       0.0       2.4000    0.0       2.0000    0.0       0.0
          0.7000    0.6000    0.7000    1.5000    2.1000    1.8000    2.0000    2.0000
          2.3000    2.6000    1.7000    0.0       0.0       0.0       0.0       0.0
          0.0       2.7000    2.5000    2.3000    2.1000    2.1000    2.0000    2.7000
          0.0       0.0       0.0       0.0       0.0       0.0       0.0       0.0
          2.4000    2.7000    1.8000    2.5000    2.2000    1.9000    2.3000    2.0000
          0.0       0.0       0.0       0.0       0.0       0.0       0.0       0.0
          3.0000    0.0       0.2000    0.2000    0.2000    0.7000    1.9000    2.5000
          0.0       0.0       0.0       3.0000    1.9000    1.4000    0.0       0.0

SMIN      1.9267
SMALL       1.9267   17.8294 10000.0000 10000.0000    32.8790

1 INPUT PATTERN CLASS,  1 OUTPUT PATTERN CLASS

SMIN      0.2341
SMALL      13.0296     0.2341 10000.0000 10000.0000    19.4861

2 INPUT PATTERN CLASS,  2 OUTPUT PATTERN CLASS

SMIN    12.8925
SMALL 10000.0000 10000.0000    12.8925    12.9077 10000.0000

3 INPUT PATTERN CLASS,  3 OUTPUT PATTERN CLASS

SMIN      0.2477
SMALL 10000.0000 10000.0000    19.7303     0.2477 10000.0000

4 INPUT PATTERN CLASS,  4 OUTPUT PATTERN CLASS

SMIN    15.1582
SMALL      25.3414   20.7925 10000.0000 10000.0000    15.1582

5 INPUT PATTERN CLASS,  5 OUTPUT PATTERN CLASS
```

REFERENCES

[1] T. Marill and D. M. Green, "Statistical recognition functions and the design of pattern recognizers," IRE Transactions on Electronic Computers, December 1960.

[2] J. S. Tsai and C. L. Sheng, "A preprocessing method to identify pattern classes with a deterministic pattern function," The University of Ottawa, Department of Electrical Engineering, Technical Report No. 67-12, November 1967.

[3] N. J. Nilsson, Learning Machines. New York: McGraw-Hill, 1965.

[4] George Nagy, "State of the art in pattern recognition," Proc. IEEE, vol. 56, pp. 836-862, May 1968.

[5] Dan Mclachlan, Jr. "Description mechanics," Information and Control 1, pp. 240-260, 1958.

[6] L. P. Horwitz and G. L. Shelton, "Pattern recognition using autocorrelation," Proc. IRE, vol. 49, pp. 175-185, January 1961.

[7] P. M. Lewis, "The characteristic selection problem in recognition systems," IRE Trans. on Information Theory, vol. IT-8, pp. 171-178; February, 1962.

[8] T. Marill and D. M. Green, "On the effectiveness of receptors in recognition systems," IEEE Trans. on Information Theory, vol. IT-9, pp. 11-17; January, 1963.

[9] Widrow, B.: Generalization and Information Storage in Networks of Adaline 'Neurons,' in Yovits, Jacobi, and Goldstein(eds.), "Self-organizing Systems--1962," pp. 435-461, Spartan Books, Washington, D. C., 1962.

[10] L. A. Kamentsky and C. N. Liu, "Computer automated design of multifont print recognition logic," IBM J. Res. and Dev., vol. 7, pp. 2-13, January 1963.

[11] C. N. Liu, "A programmed algorithm for designing multi-font character recognition logics," IEEE Trans. on Electronic Computers, vol. EC-13, pp. 586-593, October 1964.

[12] C. N. Liu and G. L. Shelton, Jr, "An experimental investigation of a mixed-font print recognition system," IEEE Trans. on Elecytonic Computers, vol. EC-15, pp. 916-925, December 1966.

[13] Rob J. Roy and James Sherman, "Two viewpoints of k-tuple pattern recognition," IEEE Trans. on Systems Science and Cybernetics, vol. SSC-3, No. 2, pp. 117-120, November 1967.

[14] R. J. Spinrad, "Machine recognition of hand printing," Information and Control 8, pp. 124-142, 1965.

[15] F. L. Alt, "Digital pattern recognition by moments," J. ACM, vol. 9, pp. 240-258, April 1962.

[16] W. Doyle, "Operations useful for similarity-invariant pattern recognition," J. ACM, vol. 9, pp. 259-267, April 1962.

[17] J. K. Brousil and D. R. Smith, "A threshold logic network for shape invariance," IEEE Trans. Electronic Computers, vol. EC-16, pp. 818-828, December 1967.

[18] Benjamin C Kuo, Automatic Control Systems. Englewood Cliffs, N. J. : Prentice-Hall, Inc., 1965.

[19] F. B. Hildebrand, Introduction to Numerical Analysis. New York: McGraw-Hill, 1956.

[20] Athanasios Papoulis, Probability, Random Variables, and Stochastic Processes. New York: McGraw-Hill, 1965.

[21] Golomb, Baumert, Easterling, Stiffler, Viterbi, Digital Communications With Space Applications. Englewood Cliffs, N. J.: Prentice-Hall, Inc., 1964.

VITA

Name:	Jo-sung Tsai
Born:	Taiwan, China, September 22, 1940.
Education:	
Primary:	Nuan-Nuan Primary School, Taiwan, China.
Secondary:	Keelung High School, Taiwan, China.
University:	Cheng Kung University, Taiwan, China. (1959-1963)
	Chiao Tung University, Taiwan, China. (1964-1966)
Degrees:	B.S.E.E. (1963)
	M.Sc. (1966)