CANADIAN THESES

THÈSES CANADIENNES

NOTICE

The quality of this microfiche is heavily dependent upon the quality of the original thesis submitted for microfilming. Every effort has been made to ensure the highest quality of reproduction possible.

If pages are missing, contact the university which granted the degree.

Some pages may have indistinct print especially if the original pages were typed with a poor typewriter ribbon or if the university sent us an inferior photocopy.

Previously copyrighted materials (journal articles, published tests, etc.) are not filmed.

Reproduction in full or in part of this film is governed by the Canadian Copyright Act, R.S.C. 1970, c. C-30. Please read the authorization forms which accompany this thesis.

AVIS

La qualité de cette microfiche dépend grandement de la qualité de la thèse soumise au microfilmage. Nous avons tout fait pour assurer une qualité supérieure de reproduction.

S'il manque des pages, veuillez communiquer avec l'université qui a conféré le grade.

La qualité d'impression de certaines pages peut laisser à désirer, surtout si les pages originales ont été dactylographiées à l'aide d'un ruban usé ou si l'université nous a fait parvenir une photocopie de qualité inférieure.

Les documents qui font déjà l'objet d'un droit d'auteur (articles de revue, examens publiés, etc.) ne sont pas microfilmés.

La reproduction, même partielle, de ce microfilm est soumise à la Loi canadienne sur le droit d'auteur, SRC 1970, c. C-30. Veuillez prendre connaissance des formules d'autorisation qui accompagnent cette thèse.

THIS DISSERTATION HAS BEEN MICROFILMED EXACTLY AS RECEIVED

LA THÈSE A ÉTÉ MICROFILMÉE TELLE QUE NOUS L'AVONS REÇUE

A Cluster-Based Classification Method for the Recognition of Unconstrained Handwritten Numerals

Tien Dung La

A Major Technical Report

in

The Department

of

Computer Science

Presented in Partial Fulfillment of the Requirements
for the Degree of Master of Computer Science at
Concordia University
Montréal, Québec, Canada

August 1985

ABSTRACT

A Cluster-Based Classification Method for the Recognition of Unconstrained Handwritten Numerals

Tien Dung La

The advancement of character recognition has led to its many applications in various fields and it has become an effective means of processing large volumes of data. There are several schemes in character recognition. However, because of the immense variety in handwritten characters, none of existing methods has proved to be reliable in recognizing totally unconstrained numerals. In human recognition, by identifying its several characteristic attributes one can deduce the unknown character from one's knowledge. The simulation of such a process through the integration of a data base and the introduction of it into the computer will certainly narrow the gap between human and machine recognition. The objective of this study is to explore the applicability of a cluster-based algorithm to recognize the totally unconstrained handwritten numerals. In our proposed scheme, the information about characters, that is different distinctive features, is organized into a data base and the combination of these features, in which one feature can complement the discriminant power of others is explored to enhance the recognition accuracy of current methods. The results indicate that the enhancement of recognition accuracy, which is about 14% for

Multi-directional Loci method used in our work, is significant.

ACKNOWLEDGEMENTS

I wish to thank Prof. C. Y. Suen for his encouragement and support as my supervisor. His advice and counsel during the preparation of this report are deeply appreciated.

I also feel grateful to Mr. Pervez Ahmed and Dr. R. Shinghal for their useful discussions and suggestions, and the US Post Office for the supply of data.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

CHAPTER I

INTRODUCTION

1.1. Why is character recognition used in high volume data processing

In the past decades, we have witnessed an explosive growth of information in almost every aspect of our daily life. In order to cope with this accelerating speed of growth, digital computers and computer networks have been developed to generate, transmit, store, access and process immense volumes of information. The traditional ways of data communication have rapidly become more and more electronic processes. Now we can have letters mailed electronically to almost anywhere in the world instantly. An entire library can be stored on magnetic disks and can be retrieved or printed out at remote sites. In banking business, the complex networks of interbanking, credit cards, automatic teller and various types of saving accounts with different daily interest rates etc., have become a reality due to the speed and precision of digital computers. With its financial rules and taxation increasingly more complex every year, the tax administration has become more and more dependent on digital computers for its operation. In industry, automation is on most planning agendas of corporations for efficiency and reliability. In research and development areas, Computer Aided Design (CAD) and

Computer Aided Manufacturing (CAM) have become indispensable tools for their versatilities and applications.

It seems that we have just entered an Information age in which man and his intelligent machines can carry out tasks much more complex and faster than ever before. The work forces have become increasingly more productive and the quality of life has been greatly improved.

However the communication between man and machine is still far from perfect. We need a more friendly man-machine interface. The current efforts in developping fifth generation computers and artificial intelligence are decisively aimed at solving this problem. Especially in the area of character recognition, during the last two decades, a lots of intensive researches have been done. It can provide a solution for processing large volumes of data automatically, e.g. source data entry, postal code reading [1], [2], [3] etc.

1.2. Advances in character recognition

Owing to intensive research in the past decades, the recognition of machine printing has become a commercial reality for more than a decade. Faced with the fact that there are hundreds of type fonts and print fonts used in the world, special fonts which facilitate processing by optical readers and enhance recognition rates have been developed. Among the most commonly known are the OCRA character set by

ANSI [4] and OCRB by European Computer Manufacturers Assocciation [5]. By Canadian standard, OCRA and OCRB character sets have been established [6], [7] together with an alphanumeric set [8]. Using these character sets, commercial OCR machines have become reliable tools for processing large volumes of bills, checks, credit card slips, medicare slips, cashier tapes and various paper forms.

Another challenging area of research is the development of reading machines for the blind, but the progress in this field has been hindered by the great number of fonts used in printings and the problem has not been completely solved. There exists commercial reading machines by various manufacturers (e.g., IBM, Control Data Corp.) for several common fonts only.

Besides the well-defined fonts and printed materials, there exists a great demand for handprinted character recognition systems. There are various possible applications, e.g., handwritten computer programs, drawings, checks, tax and medicare forms, vehicle and student registration forms, handwritten envelopes etc. Especially in the case of Zip codes, pressed by the need to process large quantities of mail, the research and development of reliable OCR techniques for mail-sorting has been carried on intensively and could be considered as one of the primary factors providing an impetus to the advancement.

However, dealing with handprinted characters is a much more difficult task for there is an infinite variation in handwritings because of a writer's own way and style of writing. Even human beings make about 4% mistakes in identifying isolate handwritten characters in the absence of context [9], [10]. It is clearly indicated that more reliable and sophisticated techniques are required for recognizing of handprints.

Significant progress has been made in this direction, a recognition accuracy of over 99% for constrained handprinted FORTRAN texts has been reported [11], [12]. In Fujimoto et al. [12] as well as R. O. Duda and P. E. Hart [13] experiments, contextual information has been employed to improve recognition. Usually the context analysis follows one of the following approaches :

- Table look-up : a table of keywords is used for references

- Use of probabilities : the probability of occurrence of character pairs (bigrams) and character triplets (trigrams) in English text is used to verify the validity of a recognized character.

Another promising area is on-line character recognition. On-line character recognition means that characters are recognized as they are written on a writing surface such as a graphic tablet or a CRT with a light pen etc. On-line

character recognition has several applications as : computer-aided design [14], and handwritten programs [15].

1.3. What is a character recognition system

The functional block diagram of a typical handprinted character recogition system is shown in Fig. 1

Input characters are read and digitized by an optical scanner. The heart of a scanner is the image sensor array, the most popular one is the charge-coupled device (CCD). Usually the sensor analog output is binarized by applying a threshold that is optimal to the system. If the input is handwritten cursively then each character is located and separated from the other characters under software control of the computer. Then some of the following preprocessing techniques can be applied on the resulting character matrix :

- smoothing : filling of holes and breaks in line segments
- elimination of noise : clean up isolated black spots
- thinning : a skeleton of character is obtained
- size and slant normalization.

Distinctive features are extracted from the preprocessed character for classification. Many different sets of features can be derived under different predefined extraction schemes. In the classification stage, usually based on extracted features and statistics of features obtained from a training set of samples or by its structural

characteristics, a classifier assigns an unknown character to its proper class. And of course, for text recognition we can apply context analysis to enhance recognition accuracy.

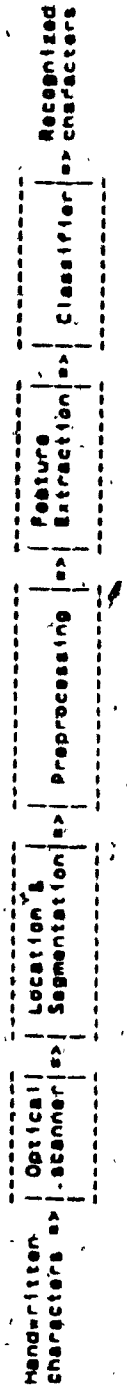


Fig. 1 Block diagram of a typical OCR system .

CHAPTER II

CHARACTER RECOGNITION METHODS.

There are two main recognition approaches, namely :

- Statistical approach.
- Syntactic approach.

2.1. Statistical approach.

In this approach, a set of characteristic attributes, called features are extracted from characters and the recognition is usually done by partitioning the feature space.

Suppose that N features are extracted from each input character and each set of N features can be considered as a feature vector X. The problem of classification is to assign each vector in the N-dimensional feature space to a proper character class. This task can be carried out by partitioning the feature space into mutually exclusive regions, where each region will correspond to a character class.

Let $C_1, C_2, \ldots, C_t$ be character classes, and

$$X = \begin{pmatrix} x_1 \\ x_2 \\ . \\ . \\ . \\ x_f \end{pmatrix}.$$

be the feature vector, where x_i is the i-th feature component, and $D_j(X)$, $j = 1, 2, \ldots, t$ denotes a function used to decide to which character class an unknown character belongs. This function is usually called discriminant function.

Then the feature vector X is in class C_i ($X \in C_i$) if

$D_i(X) > D_j(X)$, $i,j = 1, 2, \ldots, t$ $\quad i \neq j$

That is for all $X \in C_i$, $D_i(X)$ must be the largest.

In a character recognition system, sometimes the number of extracted features is quite large (for example, Chinese and Japanese characters, etc.), then the reduction of complexity can be achieved by describing the complex character as a composition of simpler subpatterns. The syntactic and structural approaches have been proposed to meet this challenge.

2.2. Syntactic approach.

For a character recognition system, the structural information is important. This information can be represented in a hierarchical manner, that is, a character is described by simpler subpatterns (primitive elements) and each subpattern can be described in terms of even simpler subpatterns. This hierarchical representation is analogous to the syntactic structure of languages and the theory of formal languages [16] can be applied to recognize characters.

For a formal language, we can define a grammar G as :

G = (V_n, V_t, P, S) where

V_n : a set of nonterminals (variables)

V_t : a set of terminals (constants)

P : set of productions or rewriting rules

S : the start or root symbol

The language generated by G, denoted by L(G), is the set of strings which satisfy two conditions :

_ each string is composed only of terminals.

_ each string can be derived from S by suitable applications of rules from the set P.

Example :

For the grammar G = (V_n, V_t, P, S) where

V_n : < S >

V_t : < a,b >

P : < S -> aSb, S -> ab >

Applying each production rule once, we obtain

S => aSb => aabb

In character recognition the features can be considered as terminals. Assume that there is a grammar such that the generated language consists of sentences (characters) which belong exclusively to one of the classes C_i ($1 \leq i \leq t$). Thus a given unknown character can be classified to belong to class C_i if and only if it is a sentence (character) of $L(G_i)$.

Some main features that can be considered as primitives :

* Straight line segments : — | / \

* Curves : ∪ ∩ ⊃ ⊂

* Loops : O

* Branches : Y X K + ⊢

In all the above mentioned approaches, the characteristic attributes, called 'features', are supposed to be invariant or less sensitive with respect to the variations and distortions of characters. Then the problem is what features should be extracted from the input character. This task is usually carried out based on either the importance of the features in characterizing characters or the contribution of features to the accuracy of recognition.

CHAPTER III

FEATURE EXTRACTION

3.1. Feature classification

The purpose of feature extraction is to find a transformation that maps the character images into a set of characteristics, called features, that contain some relevant or discriminatory information required for a recognition system.

Generally speaking, the features of a character class are the characterizing attributes of all characters which belong to that class. Due to the great variation in the handwritings, there are many possible ways to define a set of features. However, based on past research, features can be classified into the following main families (C. Y. Suen [17]) :

3.1.1. Numerical features

- Distribution of points : the measurements of the distibution of points in the character matrix which provide the positional information, density, distance of certain points from reference points as well as crossings.

- Transformation : a character matrix can be transformed into a vector, a series such as Fourier series, or a corresponding waveform.

_ Physical measurements : the respective width and height of the character can be obtained from the number of rows and columns occupied by the pattern.

3.1.2. Structural features

_ Line segments and edges : Edges and line segments are detected from the character. From the above information, such features as line lengths, and line ends can be obtained.

_ Outline of the character : The contour of a character is traced.

Contour tracing can generate the following features :

* Line tips, commonly known as terminals or end points.

* Length of line segments, including perimeter.

* Sharp angles, protrusions and spikes : > ∧ |

* Arcs, bends : ∩ ⊂

* Splits : — —

* Loops, circles : O

* Points of inflection : ∫

* Concavities and convexities : ( )

_ Center-line of the character : By applying the thinning

process on a character outline, the center-line of a character, commonly called skeleton is obtained.

From the skeleton, the following features can be obtained :

* Line tips.

* Straight line segments, horizontal and vertical.

* Diagonal lines, slants.

* Concavities.

* Loops.

* Intersections, forks, branches, nodes.

Most handwritten characters are subject to the deformations caused by defects in equipment such as writing surfaces or writing instruments and human errors. The most common factors are :

_ noise : bumps and gaps in lines, broken lines etc.

_ distortion : rounding of corners, protrusion, dilation and shrinkage, etc.

_ style variation : different shapes used to represent the same character

_ translation : movement of the whole character or its component

- rotation : change in orientation of character

In most character recognition problems which arise in practice, the determination of a complete set of discriminatory features is extremely difficult, if not impossible, because of the fact that most feature extraction methods are ad hoc in nature, due in part to our lack of understanding of the true character features and in part to the limitations of current technology.

So far none of the existing feature extraction schemes can claim to be the best or optimum. Therefore a workable character recognition machine should use a combination of methods, each of which may compensate for the shortcomings of the other.

3.2. Combination of features

An attempt has been made by Spanjersberg [18] to use a combination of different features for the recognition of handwritten digits.

In his work, Spanjersberg has employed three different schemes of feature extraction. Each scheme consists of a pre-processor for detecting features and a classifier.

The three schemes are :

- Moments

_ Glucksman [18] characteristic loci

_ Views

The moment-feature is described as

$$M_{pq} = \sum_{i=1}^{n} \sum_{j=1}^{m} z_{ij} \cdot i^p \cdot j^q$$

where

n, m : respective numbers of rows and columns of the character matrix

i, j : indicates rows and columns

z_{ij} : binary indication of condition (1 or 0 in the binary image)

The moment of up to the fifth order was used with the condition ($p + q \leq 5$). The permutations of pq, by varying p and q from 0 to 5 with the condition ($p + q \leq 5$), give a total of 21 moments such as M_{05}, M_{04},..., etc. Five of the moments are given fixed values and the 16 remaining moments are regarded as features.

Patterns from one class tend to cluster in a part of the 16-dimensional feature space. By introducing separating hyperplanes that divide the space between the centers of two clusters into two parts, a pattern is classified if each of the 9 decisions (out of 10 classes of numerals from 0 to 9) is in favor of the same class.

The view-features are obtained as the 4 outside views named as lower, upper, left and right views taken from a pattern. In this way four groups of features can be distinguished : discontinuities, slopes, end points and boundaries of 'island'. (see Fig. 2)

For each view of a pattern only one feature of each group is selected with a maximum of 6 features. Then the feature vector consists of 24 components. The classification is performed by using probability of occurrence of features. A pattern is classified into a class with the highest probability of occurrence of features.

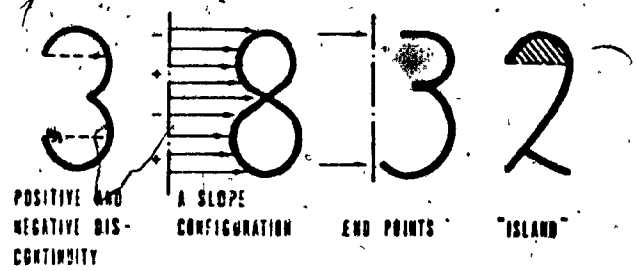


Fig. 2 Examples of features

In Glucksman's characteristic loci scheme [19], the components of feature vector are derived from the pattern's background called loci.

From a point in the pattern's background we project imaginary rays left, right, up and down. If the ray intersects a line of the pattern, the digit value assigned to that direction is 1, if more than one intersection 2 is assigned, and 0 is assigned for no intersection. Putting together these digit values, we have a four-digit code for each point of the pattern's background (see Fig. 3). And the feature space of 81 (3^4) dimensions can be defined.

In this scheme, the horizontal and vertical boundaries to the pattern form a rectangle called the characteristic rectangle and features are derived from loci inside this rectangle.

The classification is done by using a discriminant function :

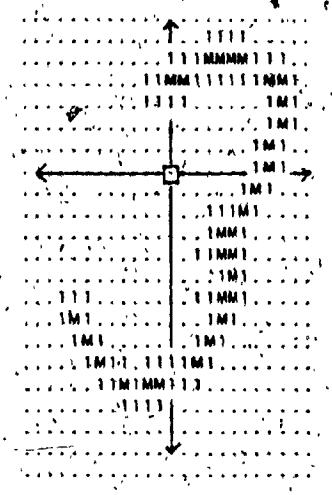
$$S = \sum_{i=0}^{81} x_i . w_i$$

where

x_i : component of feature vector.

x_0 : fixed value, e.g. +1

w_i : weight factor.



CODE 0111

Fig. 3 Characteristic loci

Spanjersberg has employed three strategies in combinations of different systems.

For combination of two different schemes, two strategies are described as :

- 'series' combination : a pattern is classified only if both systems give the same recognition.

- 'parallel' combination : a pattern is classified if both systems give the same recognition or if it is rejected by only one system.

In the third strategy, a combination of all three schemes, the correct recognition is chosen by the majority.

The results are classified into three categories : correct recognition, error and rejection. For all employed strategies, the error rates are reduced significantly from about 8% to about 2%. However only in the combination of all three schemes that the correct recognition rate is increased from about 90% to 92.8%.

3.2.1. Features used

In order to select a feasible combination of features, we start with Glucksman's characteristic loci method because this scheme has the following properties :

- The feature vector is insensitive to the translation of the pattern outline within the matrix of character

elements and to the size variation.

_ A break in the pattern outline can be tolerated if it doesn't eliminate the significant components of the feature vector.

_ It is possible to combine the characteristic loci in order to reduce the total number of components of feature vector.

_ The operations on distinct rows and columns (of the matrix of character elements) are independent, so there is a possibility for parallel processing.

A. L. Knoll [20] has done extensive experiments utilizing the Glucksman scheme for the recognition of handprinted characters on datasets from various sources such as Stanford Research Institute (SRI), Honeywell Corp., Highleyman [21]. In his experiments recognition rates for numerals varied from 74% - 84%. He observed that the recognition rate decreased considerably for unconstrained samples because of such factors as ; missing background areas, large breaks and skews in the character strokes etc.

Knoll noted that it would be useful that additional features be of the same type as the characteristic loci features so they can be extracted using the same or slightly modified logic in the software. He has also suggested that features defined from points on the character outline as

well as from diagonal directions could be used.

The Characteristic Loci resolution can be increased by using other directions such as diagonal directions. This subject has been investigated intensively by Suen [22]. The new features are called Multi-directional Loci, and one may increase the vector directions in multiples of 2, e.g., 4, 8, 16, etc.

In our work, we have tried to experiment with features defined from 'black' points (points on the character outline). However, recognizing the fact that the thickness of character strokes could affect the feature-vector, we have limited to the boundary 'black' points only for two main reasons :

- More efficient processing
- Insensitive to the thickness of character outline.

We also tried with features defined from boundary points and using four diagonal directions. However the combination of 8 directional features defined from both white points and boundary points has proved to be most effective in our experiment (see Fig. 4).

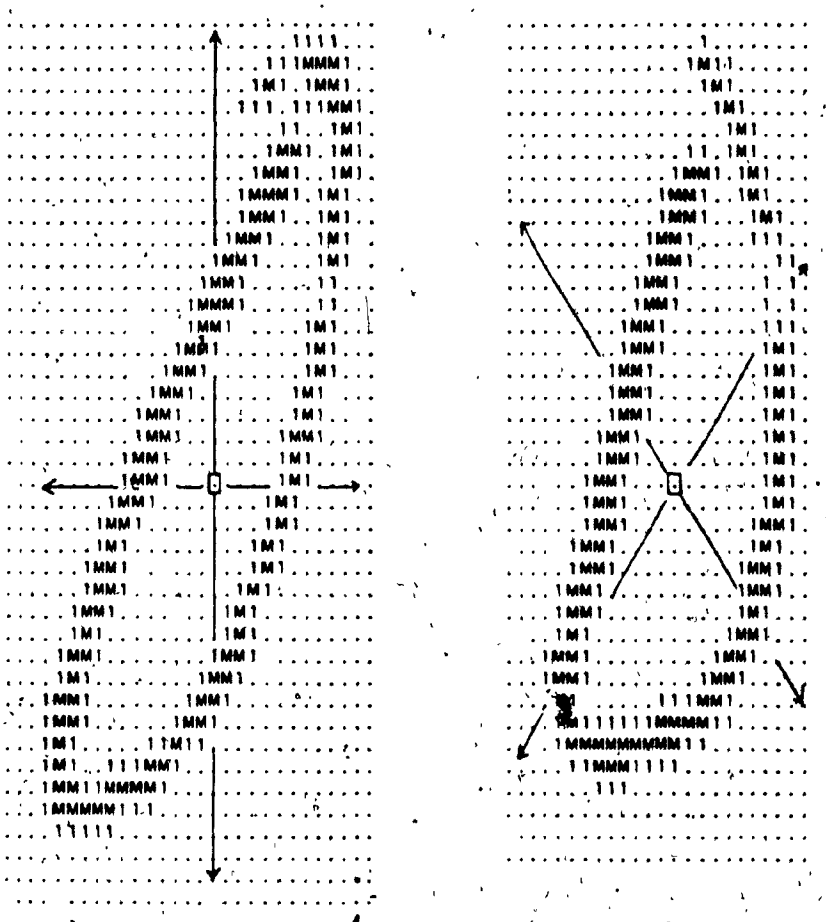


Fig. 4 Multi-directional loci

Zoning :

Some researchers have employed zoning in their feature extraction scheme. Hussain et al. [23] divided a normalized 20 X 20 character matrix into 25 distintive regions 4X4. By density of black points in these regions, a printed character can be recognized. Shridhar and Badreldin [22] have used the horizontal and vertical projections of the contours in the four quadrants of the square frame enclosing the character as features. Though these features are not sufficient for the identification of a numeral uniquely , they provide useful subclasses for further processing.

Recognizing the fact that zoning can give some useful positional information about a character, we tried to include it into the characteristic loci scheme.

In the case of handwritten numerals, there is a lot of variation in handwriting styles and irregularities as twisted lines, broken loops, etc., we find that the division of characteristic rectangle into four equal quadrants has not provided very consistent information. Instead the center of gravity of the character has a close relation with the distribution of black points. By drawing a vertical and a horizontal line through the center of gravity, we can divide the characteristic rectangle into four quadrants. The regions crossing the center of gravity, 3 rows for

horizontal direction and 1 column for vertical direction, belong to both adjacent quadrants, e.g. upper and lower quadrants or left and right quadrants.

The distributions of black points in these quadrants reflect the pattern in a more consistent way. It can give the information about the densities in the left, right, top and bottom quadrants.

With four different quadrants, we have a feature space of 81 x 4 dimensions for 4 directional loci or 81 x 8 dimensions for 8 directional loci. By the combination of features derived from both background loci and boundary points, the feature space has 81 x 4 x 2 and 81 x 8 x 2 dimensions for 4 and 8 Directional Loci respectively.

The positional information provides the distinction between left and right , top and bottom loci. It is useful in case of some numerals, for example 6 and 9.

Other features :

The center of gravity can be used not only as a reference point in zoning as described above, but from that point the enclosing character outline can be estimated grossly as described below. The coordinates of the center of gravity indicate roughly where the character outline is centered, e.g. in the upper or lower part of the chacteristic rectangle.

Then the center of gravity also provides other features which can be obtained as follows :

1_ The number of crossings in the four directions left, right, up and down originating from the center of gravity. This information can give an approximate number of horizontal and vertical lines a character has. It is useful for the discrimination between similar characters, in a sense that they have close scores in the minimum-distance classifier used in the characteristic loci method. A minimum-distance classifier computes the distance from an unknown pattern X to the prototype of each class and assigns the pattern to the class closest to it.

2 _ The center of gravity of the character can generate another feature. The coordinates of center of gravity are computed in the following way :

$$X_c = 1/T \sum_{i=1}^{n} \sum_{j=1}^{m} X_{ij} . K_{ij}$$

$$Y_c = 1/T \sum_{i=1}^{n} \sum_{j=1}^{m} Y_{ij} . K_{ij}$$

where

n, m : number of rows and columns of the character matrix

X_{ij}, Y_{ij} : X and Y coordinate of pixel P_{ij}

K_{ij} = 1 for black point

= 0 for white point

$$T = \sum_{i=1}^{n} \sum_{j=1}^{m} K_{ij} = \text{total number of black points}$$

Then new coordinates X'_c and Y'_c with respect to the characteristic rectangle are:

$$X'_c = X_c - \text{minj}$$

$$Y'_c = Y_c - \text{mini}$$

where

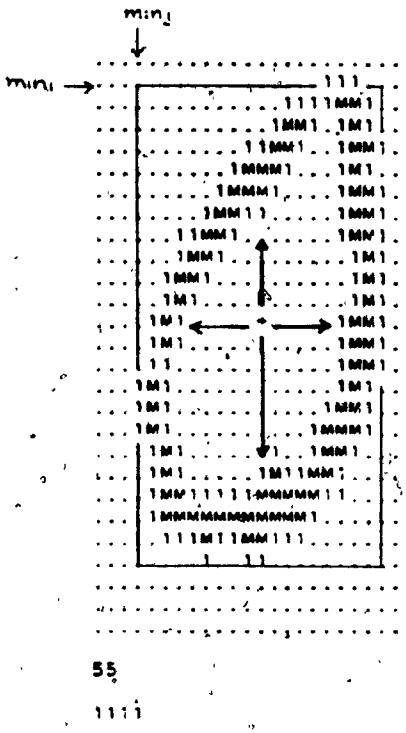
minj : indicates column number in which a black point is encountered first starting from the left of character matrix,

mini : indicates row number in which a black point is encountered first starting from the top.

Since the character matrices vary in size, then the new coordinates (X'_c & Y'_c) of the center of gravity are normalized by dividing by the width and height of the characteristic rectangle respectively and rounded to the first decimal. Putting together these first decimals, we have a code for the center of gravity.

For those examples shown in Fig. 5, with numeral 1, the center of gravity has normalized values of 0.5 and 0.5 for coordinates x and y respectively. The combination of 5 and 5 forms code 55 for the center of gravity of numeral 1. These codes reflect the symmetry and asymmetry of a character. Where codes '1111' and '0000' are resulted from the number of crossings in four directions left, right, up

and down originating from the centers of gravity of numerals 0 and 1 respectively.



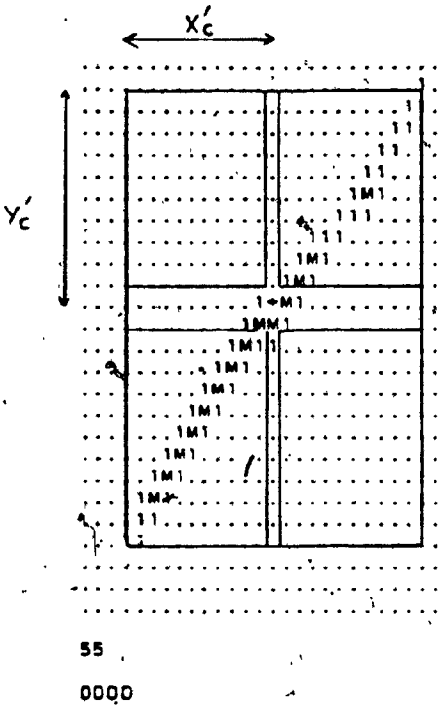


Fig. 5 Examples of feature extraction

CHAPTER IV

IMPLEMENTATION OF A CLUSTER-BASED CLASSIFICATION METHOD.

4.1. Data organization

In human recognition, a person tries to recognize a character by identifying its characteristic attributes, and recognition comes instantly. The recognition process is complicated and not well known yet. However he has to use his memory. Then he can deduce the unknown character from his knowledge about characters, different fonts, etc. acquired in his life-time. In case of confusion, mostly due to the poor handwriting, he will try to find particular features as well as contextual information that enable him to make the right distinction.

In the existing character recognition systems, usually only one set of features is employed. The basic problem is that none of the rules or schemes can cover the immense variety of handwritten characters. Then evidently, by employing existing schemes, machine recognition is usually inferior to human recognition at the current stage.

However, we have to acknowledge that the fine distinction and intelligent interpretation of the immense variety of handwritten characters by human beings can be acquired only through a lengthy learning process. The simulation of such process by the integration of a data base

which contains information such as features extracted from many different characters, etc., and the storing of this information into the computer will certainly narrow the gap between human and machine intelligence in handwriting recognition.

In our work, we made an attempt to organize the information about characters, that is features, into a data base in order to cover the great variation in handwritings. We also tried to explore the possibility of using this information to enhance the power of current recognition techniques.

Extracting features with different schemes we will get different sets of features. A classifier will make use of these features to assign an unknown character to its proper class. However in the recognition, we always have misclassification when a character is mistaken for another character.

We could consider feature extraction with a predefined scheme as viewing a character under a certain aspect of it. And certain subset of characters show similar characteristics under that aspect. However, when we change our view each member of this subset may have a quite distinct characteristic. Therefore, by combining different features obtained from different schemes, we hope to arrive at a finer distinction between characters in the process of

recognition.

In order to tackle variations in handwritings, we try to store the information about a character including its diffrent features and its proper classification in a data base for reference.

This data base is implemented by the use of an index-sequential file organization. Each record contains the information about a numeral in four fields. They are the predicted numeral which comes from Multi-directional loci scheme, the number of crossings projected from the center of gravity, the code derived from the normalized coordinates of the center of gravity and the actual identity of the numeral, respectively. The first three fields form the key for the record.

4.2. Classification method

Classifier for characteristic loci :

The pattern classification is performed by using a distance function. In our work a minimum-distance classifier is employed.

The Euclidean distance D_i between an unknown pattern X. and the i-th prototype Z_i is given by

$$D_i = \sum_{r=1}^{f} (x_r - z_{ir})^2$$

where

x_r is the r-th component of X

z_{ir} is the r-th component of Z_i

and $X \in W_i$ if $D_i < D_j$, for all $j \neq i$, for $i,j = 1,\ldots,t$

The minimum-distance classifier is most effective when patterns of each class tend to cluster about a representative pattern for that class.

Clustering :

The common problem in pattern recognition is the lack of homogeneity among attributes belonging to a pattern class. For example, in handwritten characters we can distinguish several styles used by different writers.

Different characters possess different shapes; however they can be classified into a set of prototypes. The Japanese OCR researchers have often used templates which are actually the prototypes of patterns that they wanted to recognize. Suen [20] has investigated the important role of prototypes in character recognition. Clustering techniques are employed to obtain prototypes (see Appendix B and C for examples of numeric prototypes).

Clustering techniques are used to partition a given set of entities into well-separated subsets. These subsets contain all entities that are similar according to a predefined measure of similarity.

In order to define a cluster, it is necessary first to define a measure of similarity by which patterns can be assigned to the domain of a particular cluster center. There are various typical ways to define a measure of similarity, some of them are listed in Tou et al [25], e.g.

- The Mahalanobis distance

$$D = (X - M)'C^{-1}(X - M)$$

where

M : mean feature vector

X : feature vector of unknown pattern

C : the covariance matrix of a pattern population

It is a useful measure of similarity when statistical features are considered:

_ The nonmetric measure of similarity

$$S(X,Z) = X'Z \;/\; ||X||\,||Z||$$

where

X : feature vector of unknown pattern

Y : feature vector of prototype

S(X,Z) : the cosine of the angle between the vectors X and Z.

_ Euclidean distance

It is a commonly used measure of similarity because of its simple interpretation in terms of the concept of proximity.

Once the measure of similarity has been selected the choice of a clustering criterion is very important in defining a cluster.

The clustering criterion can be categorized into two main approaches, see e.g. Tou et al [25] :

_ Heuristic scheme

_ Performance-index scheme

In the heuristic scheme, a set of rules is specified. Then based on these rules and a chosen measure of similarity, the process of clustering is performed. In this scheme the Euclidean distance is commonly used as a measure of similarity and it is necessary to set a thresnold in

order to define degrees of acceptable similarity in the clustering process.

In the performance-index scheme, a procedure is established to minimize or maximize the chosen performance-index. The most commonly used index is the sum of the square errors index (Tou et al [25]) :

$$J = \sum_{j=1}^{N_C} \sum_{X \in S_j} || X - M_j ||^2$$

where

N_C : number of clusters

S_j : denotes j-th cluster

N_j : number of samples in S_j

M_j : mean feature vector of the patterns in S_j
$= 1/N_j \sum_{X \in S_j} X$

In our work the Euclidean distance is chosen as measure of similarity for its simplicity and we employ the K-means algorithm for the clustering (Tou et al [25]).

The K-means algorithm consists of the following steps :

_ Step 1

Choose K initial cluster centers

$Z_1(1), Z_2(1), \ldots, Z_k(1)$ arbitrarily

where index 1 indicates the first iteration.

_ Step 2

At the k-th iterative step the population of samples X are distributed among K cluster domains according to the relation

$X \in S_j(k)$ if $|| X - Z_j(k) || < || X - Z_i(k) ||$, for all $i=1,2,\ldots,K$, $i \neq j$

where

$S_j(k)$ denotes the set of samples with cluster center as $Z_j(k)$, and index k indicates the k-th iteration.

_ Step 3

Compute the new cluster center $Z_j(k+1)$ at iteration k+1 such that the performance-index

$$J_j = \sum_{X \in S_j(k)} || X - Z_j(k+1) ||^2 \ , \ j=1,\ldots,K$$

is minimized. The $Z_j(k+1)$ which minimizes the performance index is simply the sample mean of $S_j(k)$ and is given by

$$Z_j(k+1) = 1/N_j \sum_{X \in S_j(k)} X, \text{ for } j=1,\ldots,K$$

where

N_j : number of samples in $S_j(k)$

_ Step 4

If $Z_j(k+1) = Z_j(k)$ for $j=1,\ldots,K$ then the final K cluster centers have converged and the procedure is terminated. Otherwise go back to step 2.

In our work, the initial value of K is set as 30 and X is a feature vector which consists of 81x8x2 components.

Then if the first two characters selected by the minimum-distance classifier belong to the same class then it is accepted as a recognized character, otherwise it is rejected and subjected to the verification process as described below.

The classification using Multi-directional loci scheme is intended to be used as a prediction and used as the first component in the key for the index-sequential file. The other two components are the number of crossings projected from the center of gravity and the code obtained from the normalized coordinates of the center of gravity with respect to the characteristic rectangle.

The index sequential file is used as a memory bank to store the information about a character including its different features and its proper classification.

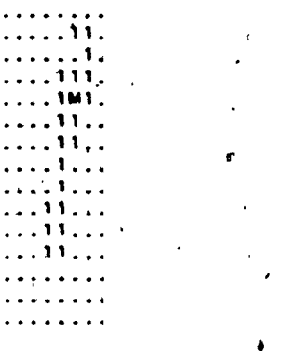
Our strategy of verification consists of the following steps :

_ Take the first predicted character and form the key to access the index sequential file. If access is successful, then the character stored in the data field of the record is considered as the true identity of unknown character and the recognition is done. Otherwise proceed to the next step.

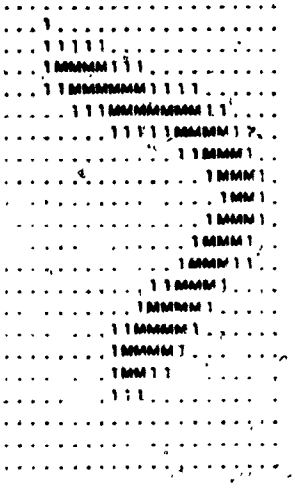
_ Take the second predicted character and form the key. If the record does exist then the unknown character is classified as mentioned in the previous step. Otherwise, the unknown character is not recognized; that is, it is rejected.

Examples of this process are shown on Fig. 6

In the character matrices, the boundary black points are indicated by digit '1'. In the first case, the numeral 1 is predicted as 7 and 1. However with additional features : '0000' and 43, where code '0000' were derived almost exclusively from numeral 1, the unknown numeral is correctly identified. In the second case, the numeral 7 is predicted as 1 and 7, but code '0111' cannot be obtained from numeral '1', then the second predicted numeral is identified as the correct one.



```
THE LIST OF MOST PROBABLE CHAR. IN ORDER AS :
- CHARACTER = 7
- CHARACTER = 1
7 0000 43
RECOGNIZED CHARACTER AS : 1
```



```
THE LIST OF MOST PROBABLE CHAR. IN ORDER AS :
- CHARACTER = 1
- CHARACTER = 7
1 0111 45
7 0111 45
RECOGNIZED CHARACTER AS : 7
```

Fig. 6 Examples of classification

CHAPTER V

EXPERIMENTATION AND RESULTS

5.1. Datasets

In our work, samples of totally unconstrained handwritten numeric Zip codes were obtained from the American Post Office. These codes were written on the envelopes by different persons from various locations in US.

The hand addressed envelopes have shown that the humans have a tendency to write well segmented numerals. However, in handwritten numerals 2, 5 and 8, the last stroke used in forming the character tended to be drawn toward the following numeral. About 85% of the Zip codes are well segmented. For the rest, the segmentation is done using analysis of connected regions [26].

Originally Zip codes on the envelopes were digitized by an optical scanner with 16-grey levels, and were binarized by choosing an optimal threshold to the system, which was established by experimentation. The digitized images were segmented into individual character matrices. Some preprocessing techniques such as the elimination of noise and smoothing were used to clean up isolated black spots, bumps and gaps in line segments. (see Appendix A for examples of samples). In order to normalize the size of the character each component of feature vector X is normalized

using the following equation :

$x'_i = x_i * 100 / Total$

where Total : total number of boundary points from which loci features are taken , or total number of white points in the characteristic rectangle when loci features are extracted from the white points. For details, see the description on page 23 and Fig. 4.

In our work, 4 and 8 Directional Loci features defined from boundary points of the character outline were considered. However the combination of 8 Directional loci features defined from both boundary point and white point (denoted by 0 in the binary image) has proved to be most effective in our experiment. The statistics of features were obtained from a training set of 1090 samples and the clustering technique was applied to form prototypes. We have tried up to 30 prototypes per numeral. Statistical values of these features were stored as reference vectors. There were 333 new samples in the testing set, and 390 samples from the training set were used for testing. For each sample three different sets of features were extracted. If a sample is rejected after the Characteristic Loci classification according to the established criteria, the cluster-based classification is applied on that sample to resolve the ambiguity. . If the verification is not successful, then that sample is not recognized.

5.2. Results

The results of recognition on training and testing datasets are listed below :

Table I Recognition Results using Multi-directional Loci

	Correct	Error	Reject
Training set	84.62 %	00.00 %	15.38 %
Testing set	71.77 %	04.20 %	24.20 %

A. Confusion Table for training set

**	0	1	2	3	4	5	6	7	8	9	Rej.	Tot.
0	59	0	0	0	0	0	0	0	0	0	1	60
1	0	40	0	0	0	0	0	0	0	0	1	41
2	0	0	49	0	0	0	0	0	0	0	18	67
3	0	0	0	49	0	0	0	0	0	0	16	65
4	0	0	0	0	6	0	0	0	0	0	6	13
5	0	0	0	0	0	26	0	0	0	0	3	29
6	0	0	0	0	0	0	23	0	0	0	1	24
7	0	0	0	0	0	0	0	40	0	0	5	45
8	0	0	0	0	0	0	0	0	23	0	5	28
9	0	0	0	0	0	0	0	0	0	15	3	18
Tot.	59	40	49	49	6	26	23	40	23	15	60	390

B. Confusion Table for testing set

**	0	1	2	3	4	5	6	7	8	9	Rej.	Tot.
0	44	0	0	0	0	0	0	0	0	0	4	48
1	0	42	0	0	0	0	0	0	0	0	0	42
2	0	0	30	0	0	0	0	0	0	0	12	44
3	0	0	0	41	0	0	0	3	1	0	34	79
4	0	0	0	0	4	0	1	3	0	0	2	10
5	0	0	0	0	0	17	0	0	0	0	10	27
6	0	0	0	0	0	0	17	0	0	0	4	24
7	0	0	0	0	0	0	0	25	0	0	2	27
8	1	0	0	0	0	0	0	0	10	2	8	21
9	0	0	0	0	0	0	0	1	0	9	4	14
Tot.	45	42	30	41	4	17	18	33	12	11	80	333

Table II Recognition Results using Cluster-based classification

	Correct	Error	Reject
Training set	99.23 %	00.77 %	00.00 %
Testing set	85.59 %	10.51 %	03.90 %

A. Confusion Table for training set

**	0	1	2	3	4	5	6	7	8	9	Rej.	Tot.
0	60	0	0	0	0	0	0	0	0	0	0	60
1	0	41	0	0	0	0	0	0	0	0	0	41
2	0	0	67	0	0	0	0	0	0	0	0	67
3	0	0	1	62	0	0	0	1	0	1	0	65
4	0	0	0	0	13	0	0	0	0	0	0	13
5	0	0	0	0	0	29	0	0	0	0	0	29
6	0	0	0	0	0	0	24	0	0	0	0	24
7	0	0	0	0	0	0	0	45	0	0	0	45
8	0	0	0	0	0	0	0	0	28	0	0	28
9	0	0	0	0	0	0	0	0	0	18	0	18
Tot.	60	41	68	62	13	29	24	46	28	19	0	390

B. Confusion Table for testing set

**	0	1	2	3	4	5	6	7	8	9	Rej.	Tot.
0	47	0	0	0	0	0	0	0	0	1	0	48
1	0	42	0	0	0	0	0	0	0	0	0	42
2	0	0	39	3	0	0	0	1	1	0	0	44
3	0	2	0	65	1	0	0	3	2	1	5	79
4	0	0	0	0	5	0	1	3	0	0	1	10
5	0	0	2	1	0	19	0	0	2	0	3	27
6	0	0	0	0	0	3	18	0	0	0	0	21
7	0	0	1	0	0	0	0	26	0	0	0	27
8	1	0	0	0	0	1	0	0	14	2	3	21
9	0	0	0	0	0	0	0	3	0	10	1	14
Tot.	48	44	42	69	6	23	19	36	19	14	13	333

Although the direct comparison of recognition results of different researchers is virtually impossible owing to the wide range of experimental procedures, constraints imposed and datasets used, we would like to list some results of various authors as below. These results are reproduced from [27] :

Table III Recognition Results of handprinted numerals

Author	Source of data	No. of samples Training	Testing	Correct (%)	Error (%)	Reject (%)
Highleyman	Highleyman	same 500		83.00		
Knoll	Highleyman	not fixed		73.70	16.50	9.80
Knoll	Munson	total = 1470		82.80	8.00	9 20
Ali et al.	Munson	480	840	92.98	3.57	3.45
Spanjersberg	Giro-document	30000	30000	92.80	2.70	4.50

CHAPTER VI

CONCLUSIONS

The cluster-based algorithm can be applied to recognize the totally unconstrained handwritten numerals. The results indicate that the combination of different features can enhance the recognition accuracy of existing methods such as the Multi-directional Loci scheme used in our experiment. However in order to arrive at a finer distinction, a systematic selection of distinctive features, in which one feature can complement the discriminant power of others, is required. The numerical features can be complemented by structural features or vice versa. By organizing the information about characters including its different characteristic attributes into a data base, the immense variety in handwritten characters could be tackled to a certain extent.

By using the Multi-directional loci scheme, we have obtained a recognition accuracy of 84.62% on the training set and 71.77% on the testing set. In another experiment, the Multi-directional loci scheme was used to predict the possible classes of an unknown numeral. In this scheme, additional features (the number of crossings and the code deriving from the center of gravity of the character) were used to recognize an unknown as one of the predicted classes. Using this scheme, 99.23% of the training set

samples were correctly recognized; whereas, on the testing set samples, 85.59% correct recognition was achieved. The enhancement of about 14% in recognition accuracy was obtained due to the usage of additional features.

REFERENCES

1. Focht, L. R., and A. Burger, " A numeric script recognition for postal zip code application," Proc. Int. Conf. Cybernetics and Society, pp. 489 - 492, 1976.

2. Genchi, H., et al, " Recognition of handwritten numeral character for automatic letter sorting," Proc. IEEE, vol. 56, pp. 1292-1301, Aug. 1968

3. Genchi, H., et al, " Automatic reader-sorter for mail with handwritten or printed postal code numbers," Toshiba Rev., pp. 7-11, July-Aug. 1970

4. " Character Set and Print Quality for Optical Character Recognition (OCR-A)," American National Standards Institute, 1977.

5. " The Alphanumeric Character Set for OCR-B for Optical Recognition," European Computer Manufacturers Association, 1971.

6. " Style A Character Set for Optical Character Recognition," Canadian Standards Association, 1975.

7. " Style B Character Set for Optical Character Recognition," Canadian Standards Association, 1975.

8. " Alphanumeric Character Set for Handprinting," Canadian Standards Association, July 1983.

9. Suen, C. Y., et al, " Dispersion factor : A quantitative measurement of the quality of handprinted characters," Proc. Int. Conf. Cybernetics and Society, pp. 681-685, Sept. 1977.

10. Suen, C. Y., and R. J. Shillman, " Low Error Rate Optical Character Recognition of Unconstrained Handprinted Letters Based on Model of Human Perception," IEEE Trans. Syst., Man, and Cybernetics, vol. 7, pp. 491-495, June 1977.

11. Lin, W. C., and T. L. Scully, " Computer Identification of constrained handprinted characters with a high recognition rate," IEEE Trans. Systems, Man, and Cybernetics, vol. 4, pp. 497-504, 1974.

12. Fujimoto, Y., et al, " Recognition of handprinted characters by non-linear elastic matching," Proc. 3rd Int. Joint Conf. Pattern Recognition, pp. 113-118, 1976.

13. Duda, R. O., and P. E. Hart, " Experiments in the recognition of handprinted texts : Part II Context analysis," Proc. Fall Joint Comp. Conf., vol. 23, pp. 1139-1149, 1968.

14. Hosaka, M., and K. Fumihiko, " An interactive geometrical design system with handwriting," Proc. Int. Fed. Info: Processing Soc. Congr., pp. 167-172, 1977.

15. Grewal, K., and J. D. Patterson, " A binary feature extraction technique," IEEE Trans. Comput., vol. 23, pp. 545-549, May 1974.

16. Chomsky, N., " Three models for the description of language," IEEE Trans. Information Theory IT-2, pp. 113-124, 1956.

17. Suen, C. Y., " Distinctive features in automatic recognition of handprinted characters," Signal Processing, vol. 4, pp. 193-207, April 1982.

18. Spanjersberg, A. A., " Combinations of different systems for the recognition of handwritten digits," Proc. 2nd Int. Jt. Conf. Pattern Recognition, pp. 208-209, 1974.

19. Glucksman, H. A., " Classification of mixed-font alphabetics by characteristic loci," Digest of 1st Ann. IEEE Comp. Conf., pp. 138-141, Sept. 1967.

20. Knoll, A. L., " Experiments with Characteristic Loci for recognition of handprinted characters," IEEE Trans. Comput., vol. 18, pp. 366-372, 1969.

21. Highleyman, W. H., " Data for character recognition studies," IEEE Trans. Electronic Computers (Correspondence), vol. EC.12, pp. 135-136, April 1963.

22. Suen, C. Y., " The role of multi-directional loci and clustering in reliable recognition of characters,"Proc. of the 6th International Conf. on Pattern Recognition, Munich, Germany, pp. 1020-1022, Oct. 1982.

23. Hussain, A. B. S., et al," Results obtained using a simple character recognition procedure on Munson's handprinted data," IEEE Trans. Comput., vol. 21, pp. 201-205, 1972.

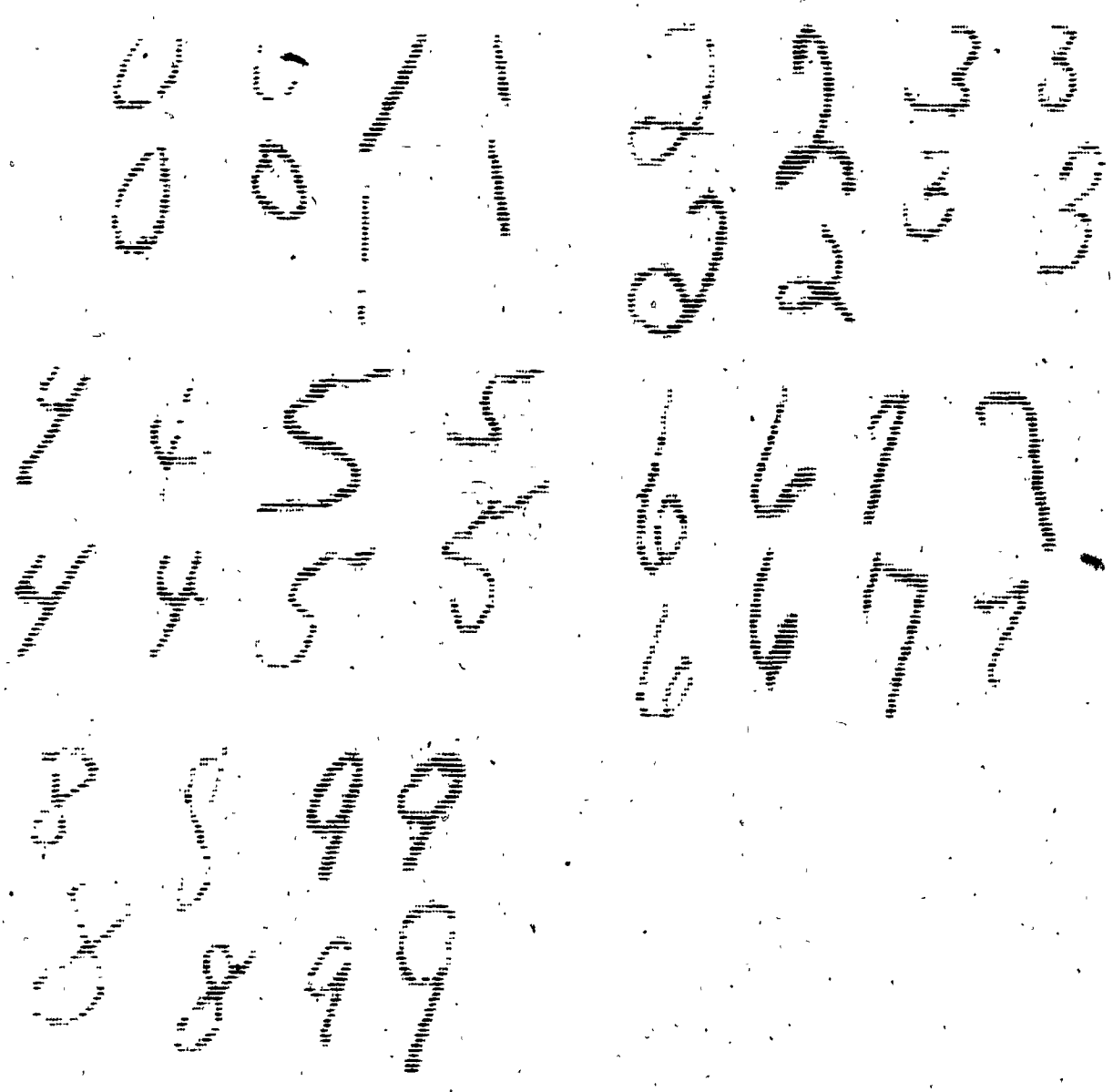
24. Shridar, M., and A. Badreldin, " Handwritten numeral recognition by tree classification methods," Image and Vision Computing, vol. 3, no. 3, pp. 143-149, Aug. 1984.

25. Tou, J. T., and Gonzalez, R. C., Pattern Recognition Principles, Addison-Wesley Publishing Company Inc., 1974.

26. Ahmed P., and Suen C. Y., " Segmentation of unconstrained handwritten numeric postal Zip codes," in Proc. of 6th Int. Conf. on Pattern Recognition, Munich, Germany, pp. 545-547, Oct. 1982.

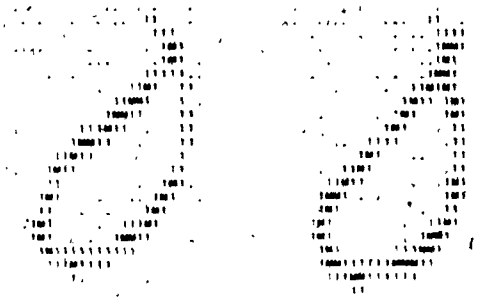
27. Suen, C. Y. et al, " Automatic recognition of handprinted characters - The state of the art," Proceeding of the IEEE, vol. 68, no. 4, pp. 469-487, April 1980.

APPENDIX A. Typical samples

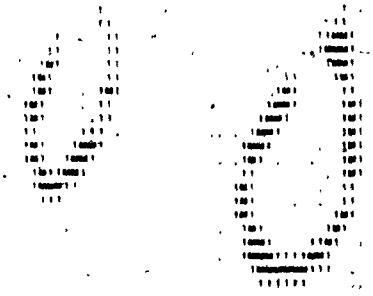


APPENDIX B. Numeric prototypes

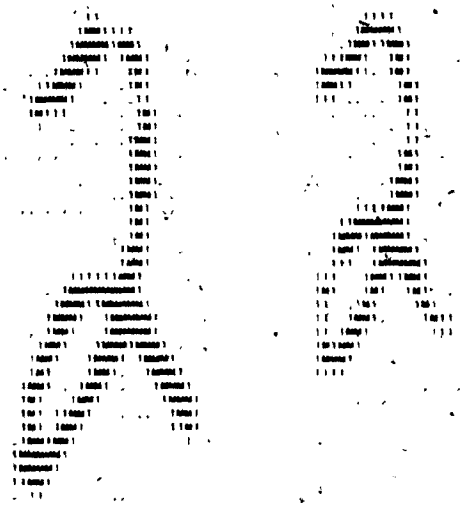
Prototype 01



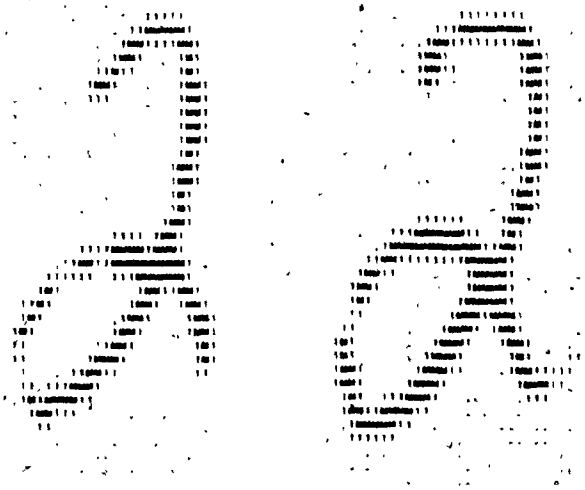
Prototype 02



Prototype 03



Prototype 21



Prototype 22

APPENDIX C. Prototypes of numeral 3