Computer Recognition of Totally Unconstrained Handwritten ZIP Codes

Pervez Ahmed

A Thesis

in

The Department

of

Computer Science

Presented in Partial Fulfillment of the Requirements
for the Degree of Doctor of Philosophy at
Concordia University
Montréal, Québec, Canada

July 1986

ISBN 0-315-32232-2

ABSTRACT

Computer Recognition of Totally Unconstrained Handwritten ZIP Codes

Pervez Ahmed, Ph.D.
Concordia University, 1986

Techniques for the recognition of totally unconstrained handwritten numerals are developed and tested on a large set (8540 numerals) of real-life Postal ZIP code samples. Different approaches, including statistical, structural and hybrid are developed and their performance measured.

The use of extended characteristic loci codes with statistical approach as a front-end recognizer and predictor of an unknown character has been investigated. The expected prediction capability of this approach lies in the interval (90%, 100%). Its recognition reliability ranges in the interval (88.55%, 89.02%) with substitution and rejection rates between (10.40%, 10.85%) and (5.23%, 5.32%) respectively.

A new approach to define and extract structural features based on edge classification has been implemented. Shapes, such as holes, simple cavities and end points, are used as structural features and the relational data-model is used as a tool to organize the information related to each shape.

Furthermore, a fuzzy set theoretic technique has been developed for the identification of unknown characters. The structural approach is used as a back-end recognizer, and the unknown characters are identified by incorporating the predicted information as well as the structural identification score. Its recognition reliability lies in the interval (99.06%, 99.8%) with substitution and rejection rates between (0.18%, 0.78%) and (7.13%, 22.48%) respectively.

In the hybrid approach, unknown characters are identified either by the statistical approach or structural approach. Characters with strong predictions (normal characters) are recognized by the statistical approach and those with weak predictions (abnormal characters) by the structural approach. Using this approach, one of the many possible combinations of the statistical and structural recognition schemes has been investigated, and its recognition reliability improved to the interval (95.94%, 96.29%) with substitution and rejection rates between (3.45%, 3.96%) and (2.36%, 7.01%) respectively.

- Image processing (enhancement, binarization and segmentation) techniques were also developed to obtain suitable binary characters from the raw images of the ZIP codes.

The techniques, image processing together with recognition, described in this thesis present a complete model of an optical reader for ZIP code recognition. The model, in general, is applicable to the text recognition problem.

Acknowledgements

I am grateful to the almighty God for granting me the wisdom, curiosity and dedication to conduct research.

I am thankful to my supervisor Dr. C. Y. Suen for his support, encouragement and guidance without which I could not have finished this work. I am pleased to acknowledge the continued encouragement given by Dr. Pankaj Goyal and Dr. Shunji Mori of Japan and am thankful for their involvement in the theoretical formulation of this work.

I am indebted to my friends Mr. Tien Dung La, Mr. T. S. Naraynan, Mr. A. K. Nanda and Mr. George Balouxis for helping me in preparing the test data and monitoring the compilation of the test results. I am especially thankful to my wife Shama for her patience.

I would like to thank Ms. Halina Monkiewicz and Mrs. Patricia Kierans for their assistance in preparing the diagrams, and Mr. Jose Suarez, Ms. Iona Farell and Mr. Daniel Soviak for their assistance in running the computer programs. I wish to thank Mr. Harold C. Glass, Manager, pattern recognition committee of the U. S. postal services for supplying the test data.

This work was supported by the research grants from the Natural Sciences and Engineering Research Council of Canada and Department of Education of Quebec.

List of Contents

Contents	Page Number

Chapter 3
Feature Definition, Extraction and Knowledge Organization

Chapter 4
Decision Making Processes in Pattern Recognition

Chapter 5
A Multi-stage Classification Scheme

Chapter 6
Test Data and Experimental Results

Chapter 7
Performance Analysis, Future Work and Conclusion

Appendix A
Confusion Tables

Appendix B
Samples From The Data Set

Appendix C
Recognition Process

List of Figures

List of Tables

Chapter 1

Introduction

The need of an interconnected society with instant access to vast amounts of data is one of the challenges of our modern society. This need can only be fulfilled if all data that already exist and that are being generated using tools, such as typewriters and pens on a non-electronic storage medium are transcribed to some electronic storage media. This has created demands for new techniques to generate, acquire, transmit, store, access, and present symbols that represent information. From the onset of data processing, the design of an efficient tool to capture data at its source has been a topic of extensive research. Existence of a variety of data entry devices are the result of this ongoing effort. The design of efficient reading machines is also an effort in the same direction. These machines are required to transcribe existing 'non-electronic' form of data onto electronic media. Examples of important existing data include books, research journals, records etc. The reading ability of machines not merely provides a solution to the data entry problems, it also gives a very flexible and economic means to use computers to solve current real-life problems, such as automatic mail sorting, processing of cheques, account sheets, transaction statements and many more business and scientific applications (Chattan and Teacher [1962]; Focht and Burger [1976]). Current advances in office automation and rising data entry costs have created more than ever the need to design a reliable and efficient reading machine. Applications of such machines in offices of tomorrow, where office workers will be equipped with work-stations (a single multiple-purpose terminal) cannot be avoided. Furthermore, the cost effectiveness of these machines in such environments is very well realized in a survey entitled "OCR In the Age of Instant Response" presented in Schantz [1982], which states:

> "... The cost to process one million bytes of information had declined sharply, from about $40 to only four cents. The cost for storing one million bytes of information for one month had dropped from $64 to $1.80. And, data processing hardware cost had dipped by more than 15 percent since 1977. However, also since 1977, data entry processing costs had also increased by approximately 14 percent yearly, and the cost to manually enter one million characters of information had risen over the past twenty years from $300 to more than $650."

Design of a reading machine requires knowledge from the areas of pattern recognition, image processing, and hardware design. The basic components of a reading machine are shown in Fig 1.1. It consists of an optical scanner, a preprocessing module, a feature extraction module and a classification module. Using the optical scanner the document to be read is scanned and a digital image of the document is sent to the preprocessing module. Functions of this module are image quality improvement and segmentation (extraction of images of individual characters). Images of the segmented characters are sent to the feature extraction module where features useful for classification are extracted. Using these features and some classification procedure, individual characters are recognized or rejected. Over the past three decades considerable attention has been paid to this challenging problem with each module being a research subject itself. Researchers from various disciplines, such as engineering, mathematics and computer science are continuously trying to provide better solutions to the problems arising in each module. Continued enhancement of the capabilities of each module has made possible the development of reading machines with limited capabilities. The historical development of some of these machines, covering the period from 1800 to 1980 are presented in Schantz [1982]. An exhaustive survey covering major recognition techniques, their comparative performance analysis and recommendations for future research directions can be found in Suen et al [1980].

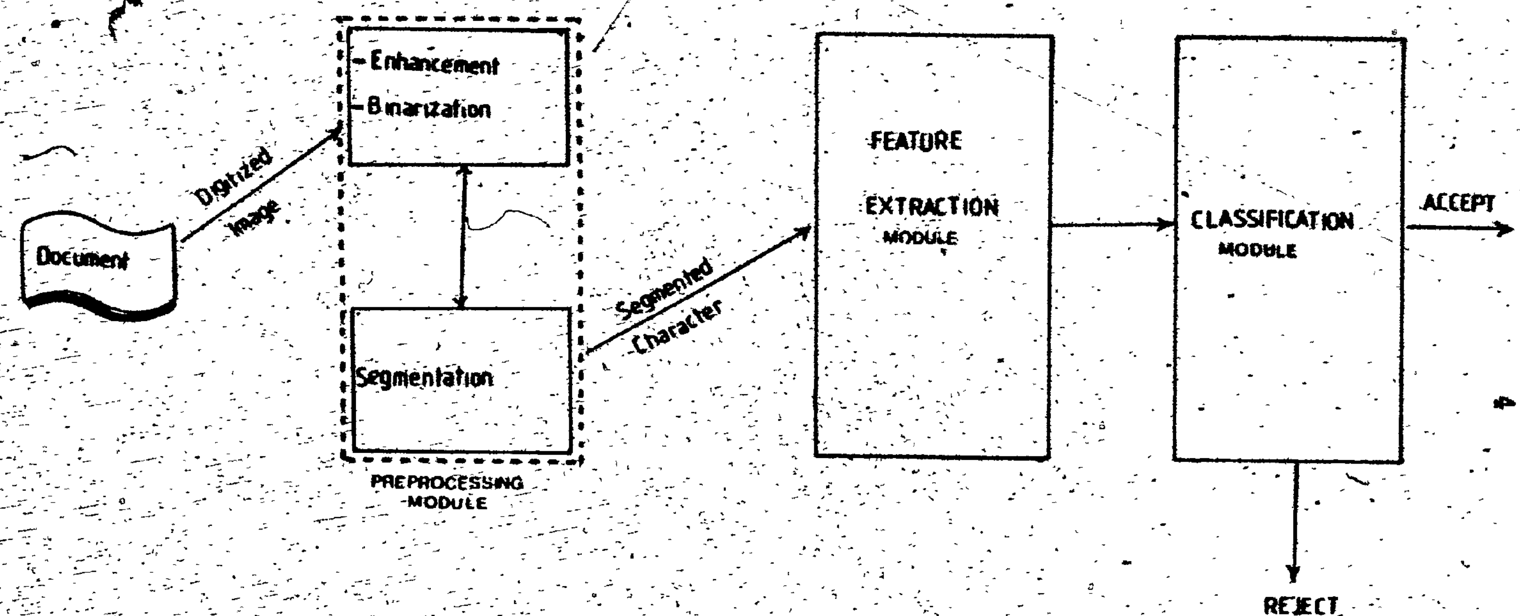


Fig. 1.1. Basic Components of a Reading Machine.

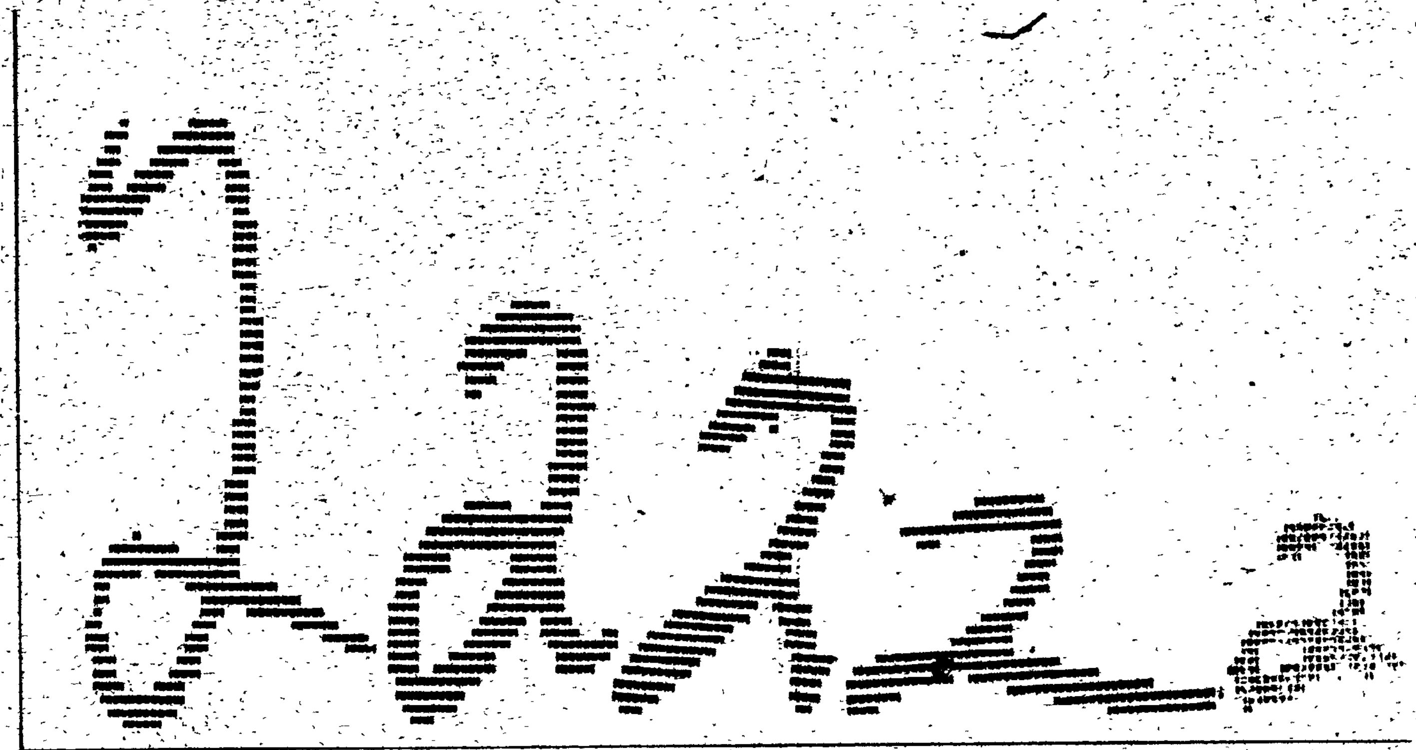


Fig. 1.2. Within Class Variations.

1.1 Objectives of this Research.

The prime objective of this research is the development of a model of a machine capable of reading totally unconstrained handwritten numerals. Observed variations in shapes and sizes established the need for the development of a generalized structural feature extraction scheme and a two stage (prediction and structural) classification scheme presented in Chapters three and five respectively. It should be noted that unlike the recognition of type-written or constrained handwritten characters, this is not just a 10 class recognition problem. Within a class, sample to sample variations in shapes, sizes, thickness and orientations etc., make this problem very complex and challenging (See Fig. 1.2 and Appendix B).

1.2 Proposed Recognition Scheme

Character recognition (in general pattern recognition) schemes can be characterized on the basis of the type of features and classification methods used. Features are generally grouped into the statistical (numerical or quantitative) and the structural (qualitative, i.e., representing the structure of patterns) groups and widely used classification methods are mathematical, statistical, grammatical, graph-theoretic and heuristic. Generally speaking, numerical features are used either in the mathematical or statistical methods and the structural features in other methods. Experience shows that schemes based on either numerical or structural features may not provide an adequate solution to problems in real-life applications such as the recognition of totally unconstrained handwritten characters or symbols (Kanal and Chandrasekran [1972]; Duerr et al [1980]). In this research, based on new statistical and structural features, recognition schemes are developed and their perfomances are analyzed. Based on the performance analysis a method to combine these schemes is explored.

In this study, statistical features are primarily used to predict the identity of an unknown character and this is referred to as the prediction module in subsequent discussions. The need for such a module arose, owing to the phenomenon of infinite variety of characters, for example a line can be drawn on a surface in infinite ways. Similarly, in an unconstrained environment, a character can be formed in many ways, thus, it is virtually impossible to enumerate (manually) the number of subclasses within a class. This module identifies an unknown pattern as a member of some probable group and provides a basis for obtaining the subclasses in the context of the predictions. For example, a sample belonging to a class, say "x", uniquely predicted as "x" is considered as a normal (completely different from other classes) sample. Samples having different predictions are considered as dubious or abnormal samples. The degree of abnormality is judged by the number of varying predictions and provides useful information for the quality assesment of the underlying patterns. The performance of this module involving different cases of predictions, such as an unknown pattern belonging to a unique class or more than one class (maximum five classes), is studied. Patterns predicted as a member of more than five classes are considered as bad samples and rejected.

The structural features representing some fundamental shapes such as cavities with different characteristics, holes and end points occurring in different locations are used in the structural scheme. A new approach to define and extract these features is developed. The relational database model is used as a tool to organize information contributed by these fundamental shapes. Based on the fuzzy set theory (Zadeh [1965]; Deluca and Termini [1972]; Deluca and Capocelli [1973]; Kaufmann [1975]) a classification scheme is implemented. The salient feature of this scheme is the concept of ideal and nonideal prototypes which can be dynamically defined on the basis of the shape information extracted from the unknown pattern and its relevance towards a class obtained from the database. A function of the fuzzy distances of an unknown pattern from the ideal and nonideal prototypes is used to compute the characteristic membership. Using the characteristic membership and predictions, patterns are classified. In subsequent discussions this scheme is referred to as the structural module.

Generally speaking, compared to structural features, the extraction of statistical features is easy and fast. However, in the presence of distortions, schemes based on the statistical features are less reliable than the schemes based on the structural features. In addition, it has been experienced that structural analysis on the normal samples (samples with high predictions) is redundant. A scheme termed as the hybrid scheme is studied. In the hybrid scheme, samples with strong predictions are recognized in the prediction module itself and samples with weak predictions are processed by the structural module. The advantage of this scheme is that the normal samples can be processed at a higher speed.

1.3 Outline of the Thesis

Except the design of scanning devices, this thesis addresses the problems faced by every module of a reading machine operating in a real-life environment. Since each module carries substantial independent research emphasis, the literature survey related to each module is included along with the relevant chapters. Preprocessing techniques, namely binarization and segmentation developed for this research are presented in Chapter two. A new approach to define and extract the structural features and modified approach to extract the characteristic loci feature, termed as the statistical feature are presented in chapter three. Classification techniques are characterized in Chapter four and proposed classification techniques, namely prediction and structural modules are discussed in Chapter five. Findings of this study and information about the test data are presented in Chapter six. Finally, the performance of the system and extension of this study for future research are examined in Chapter seven.

1.4 Test Data

The proposed recognition scheme was tested on totally unconstrained handwritten ZIP code data collected from the dead letter envelopes by the U. S. postal department at different locations was chosen. This data set reflects the true state of nature, and has been very useful in understanding problems faced by a reading machine designed to read and sort mail.

Chapter 2

Preprocessing

2.1 Introduction

Preprocessing is an important part of a pattern recognition system. The performance and reliability of the whole recognition system heavily depend on the quality of the preprocessed image. During the past few decades lots of research work in character recognition were centered around the development of good feature extraction, feature selection and classification methods. Generally speaking, some researchers in character recognition applications assume the existence of a neat, clean and well-defined pattern--the preprocessing steps are either ignored or merely discussed to the extent of the "salt-pepper" (random) noise removal. Somehow this problem was considered as a part of digital image processing. In fact, the development of a recognition system to solve the real-life problems, such as the design of a digital document processing system and machine sorting of handwritten envelopes etc. needs an efficient preprocessor to meet the real world standards.

In this research multi-gray valued digital ZIP code images are used. The preprocessing steps required to represent the constituents of each ZIP code image in a form suitable to the recognition stage are: binarization, noise cleaning and segmentation. A detailed description of the techniques implemented in this study is presented in the following section.

2.2 Binarization

The purpose of binarization is to transform a multi-gray level image into a two level image where one level represents the background region and the other level represents the object region. Thus, the binarization process involves the selection of a gray level threshold "t" such that all gray levels greater than "t" are mapped into the "object" label (generally denoted by 1), and all others less than "t" are mapped into the "background" label (denoted by 0). Techniques used to select the appropriate value of "t" are called "threshold selection techniques" in image processing.

In a survey on thresholding techniques, Weszka [1978] defined a threshold operator as a function "T" of the form

$$T(x, y, N(x,y), g(x,y))$$

where $g(x,y)$ is the gray level of the point with co-ordinates (x,y), and N(x,y) denotes some local property of the point (x,y), such as average gray level over some neighborhood. For each point (x,y) in an image, if g(x,y) > T(x, y, N(x,y), g(x,y)) then the point (x,y) is labeled as an object point; otherwise, the point (x,y) is labeled as a background point. Thresholding techniques generally fall into three categories: (1) global--where T depends only on g(x,y); (2) local--where "T" depends on both g(x,y) and N(x,y); and (3) dynamic--where "T" depends on the coordinate values x, y as well as on g(x,y) and N(x,y). Each of these techniques is briefly discussed below.

The global techniques require an analysis of the image's gray level histogram. The simplest global technique is a "p-tile" method which arbitrarily puts q% of the gray level into the object label. This method is not applicable to those cases where the object area is unknown or varies from image-to-image. Variations of this method and some others like "mode" methods which choose the threshold at the valleys of the histogram can be seen in (Pal et al [1983]; Pal and King [1981]; Pun [1980]).

The "mode" method depends on the existence of the valleys which in turn depends on the distribution of gray levels in an image. Experience shows that often histograms are unimodal or the valleys are insignificant. Therefore, histogram modification methods were developed to modify the original histogram into some desired shape (Gonzalez and Paul [1977]) and use of local statistics is explored to obtain the local thresholds (Rosenfeld and Smith [1981]; Lee [1980]; Panda [1978]).

In some applications where gray-level histograms are unimodal or the valleys are insignificant, a single global threshold may not satisfactorily work. So dynamic threshold selection techniques were developed. In dynamic threshold selection techniques the histograms of different regions are used to select the appropriate threshold for the corresponding regions in an image (White and Rohrer [1983]).

In applications such as ZIP code processing (same is true for digital document processing), where the gray level distribution varies from image to image and within image, a single global threshold hardly produces an acceptable binarized image. To overcome this a method which incorporates the local as well as global statistics has been developed in this research. This method is formally described in the next section.

2.2.1 A Region Growing Approach to Threshold Selection

An iterative scheme has been implemented to determine a binarization threshold which produces an unbroken character (upto some extent) by converting a unimodal histogram into a bimodal histogram. The number of iterations is controlled by computing the number of objects (discussed in the next section) in each ZIP code image. The basic idea behind this approach is based on the region growing process (Rosenfeld and Kak [1976]).

A ZIP code image is considered as a matrix $G(M,N)$, $M,N>1$, whose row (M) and column (N) indices identify a point (pixel) in a ZIP code image and the corresponding matrix element value $g_{i,j}$ identifies the gray level at the point (i,j). For every image the histogram of gray level is obtained by determining the frequency of each gray level. For a bimodal histogram the global threshold is determined at the valley position and for the unimodal case the global threshold is determined by the "p-tile" method. Using this global threshold, an image is binarized and passed to a segmentation process which counts the number of objects (dark regions) in an image. If the total number of objects is equal to e.g. 5 (expected number of digits in a ZIP code) then the image is considered as an acceptable image; otherwise, the following histogram modification process is invoked.

In this process the image is scanned using a nonoverlapping window of size m x n, (m,n=5) and a measure of uniformity "Q" defined below is computed for each window area.

$$Q = 1 / (1+\sigma^2)$$

where σ^2 is the gray level variance in a window. Clearly, the value of "Q" approaches to 1 for a uniform region and less than 1 for a nonuniform region. For every nonuniform region the minimum (g_{min}) and the maximum (g_{max}) gray values in a window are determined to compute the quantities

$$D_{min} = \sum_{i=1}^{m} \sum_{j=1}^{n} (g_{i,j} - g_{min})$$

$$D_{max} = \sum_{i=1}^{m} \sum_{j=1}^{n} (g_{max} - g_{i,j})$$

If $D_{min} \leq D_{max}$ then the average gray value inside the window is calculated and assigned to each element in the window; otherwise, the value of g_{max} is assigned to each element.

After each iteration, the binarized image is passed to the segmentation process, if a significant improvement (reduced number of objects) is found then the image is submitted for the next iteration; otherwise the ZIP code image is discarded. The performance of this method is shown in Fig 2.1(a-d). The gray level distribution of the original image and images obtained in the subsequent iterations are given in the following table. The original image has a linear gray level distribution (iteration "0"), after some iterations a significant valley is formed around the gray level 13, thus it is selected as the binarization threshold for this image.

Grey-levels
Dark to light

	10	11	12	13	14	
Iter. 0 !	15	214	485	1366	12283	!
Iter. 5 !		300	906	895	12271	!
Iter. 10 !		348	928	807	12253	!
Iter. 15 !		363	931	798	12244	!

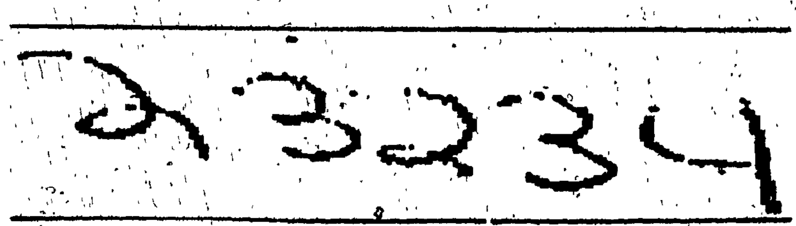


Fig. 2.1 (a). Binarization by p-tile Method.

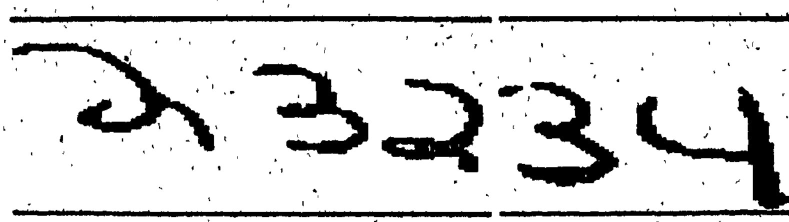


Fig. 2.1 (b). Binarization after 5th iteration.

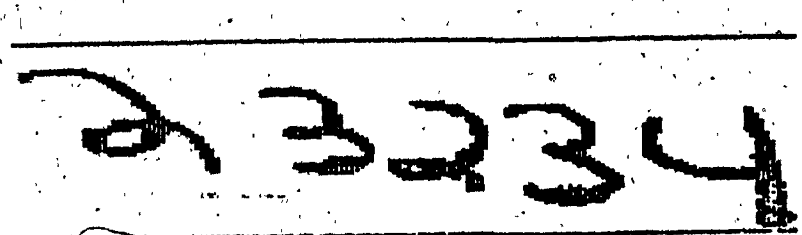


Fig. 2.1 (c). Binarization after 10th iteration.

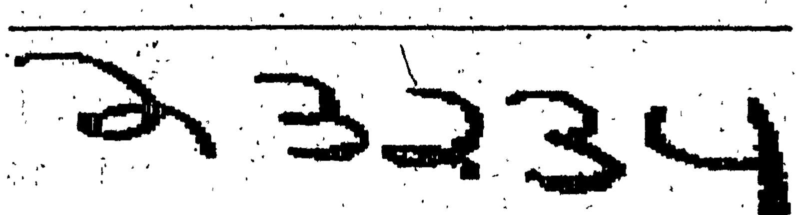


Fig. 2.1 (d). Binarization after 15th iteration.

2.3 Segmentation

Segmentation is a process to determine the constituents of an image. In digital document processing, segmentation is used to extract the image of the individual character. Like binarization, not enough attention (generally test data are generated in labs where writers were asked to write well isolated characters) is paid to this basic problem and occasionally some segmentation methods along with the details of recognition systems are discussed. These methods can be roughly categorized as "projection-based", "pitch-based", "recognition-based" and "region-based" techniques. In the "projection-based" methods the vertical projection profile of dark points of each line of the text on a horizontal line is analyzed and it is assumed that each character image lies between two consecutive minima. The "pitch-based" techniques extract the character images simply by computing the width of consecutive characters. These two techniques are suitable for type-written texts where characters are equally spaced and there is a significant gap between adjacent characters (Buldreay and Milbrandt [1962]; Soltran [1957]; Hoffman and McCullough [1971]; Mivazaki and Hoshino [1974]) In the "recognition-based" methods segmentation is performed by recognizing a character in a sequential scan (Hennis [1968]). All these methods perform satisfactorily, if the images can be isolated easily, which is true in many typewritten texts. For handwritten or handprinted texts where variations in handwritings (Suen [1973], [1974]) are unpredictable, the performance of these methods is dubious. The "region-based" method is the only alternative for the segmentation of totally unconstrained handwritten characters. In this method the different regions belonging to the character images are identified and recorded. There are several possible approaches (contour tracing, sequential scanning etc.) for object identification. In this research, a region-based segmentation method has been developed. The objective of this segmentation method is not merely to get the isolated images of the constituents of a ZIP code, but also to reveal some characteristics (touching, broken, overlapping, part of a character nested under one another but not touching, etc., (Figs.2.2(a-c))), useful in assessing the possibility of processing a ZIP code. The performance (in terms of speed and reliability) of this method has been found to be satisfactory (Ahmed and Suen [1982]). This method is based on sequential scanning. Every ZIP code image is scanned vertically from left to right and bottom up. Information gathered in two consecutive scans is used in the segmentation process.

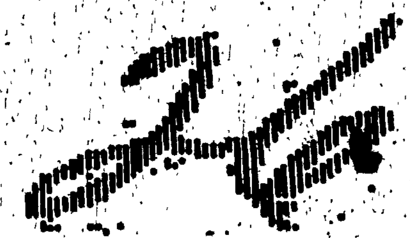


Fig. 2.2 (a). Touching Characters.

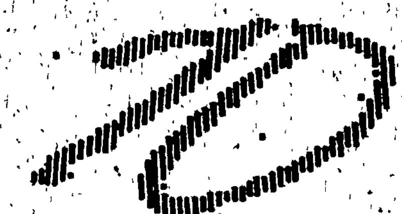


Fig. 2.2 (b). Part of a Character Nested Under Another One But Not Touching

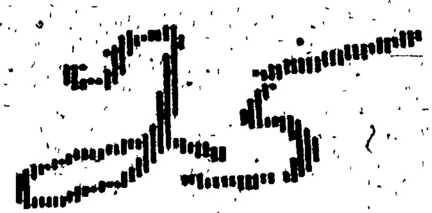


Fig. 2.2 (c). Broken Characters.

2.3.1 Preliminary Definitions

Some basic definitions and terminologies used in the explanation of the segmentation method developed in this research are discussed in this section.

Binary Image

A binarized image is a matrix $G(M,N)$ $(M,N > 1)$ where each element $g_{i,j} \epsilon \{\alpha, 1-\alpha\}$. Every binarized image consists of background and object regions. The background region is represented by $\alpha=0$ and object region by $\alpha = 1$.

Pixel Connectivity

Any two pixels in an image are said to be connected, if and only if they are 4-connected. Our experience is that 8-connectivity produces touching characters.

Components

A component is a connected set of pixels with a value α or $1-\alpha$ in an image $G(M,N)$. The k^{th} component in the i^{th} scan is defined as

$$K_i^k = \{ g_{i,j} \mid g_{i,\ell} = g_{i,\ell+1} = \ldots = g_{i,r} = \alpha, \text{ for all } j \in [\ell, \ell+1, \ldots, r] \text{ and } g_{i,\ell-1} = g_{i,r+1} = 1-\alpha \text{ for all } j \in [\ell-1, r+1] \}$$

where (i,ℓ) and (i,r) denote the starting and ending coordinates of the k^{th} component respectively.

Component Connectivity

Two components K_i^k and $K_{i-1}^{k'}$ in scan i and (i-1) with starting and ending coordinates (i,ℓ), (i,r) and $(i-1,\ell')$, $(i-1,r')$ respectively are said to be connected if any one of the following situations occurs:

(a) $\ell' \leq \ell \leq r'$ (b) $\ell \leq \ell' \leq r$

(c) $\ell \leq \ell'$ and $r' \leq r$ (d) $\ell' \leq \ell$ and $r \leq r'$

Connectivity Matrix

Let there be m and n components in the i^{th} and $(i-1)^{th}$ scans respectively. A connectivity matrix referred to as C-matrix and denoted by C(m,n) is defined as $c_{k,k'} = 1$, if the k^{th} component from the i^{th} scan is connected to the k'^{th} component of the $(i-1)^{th}$ scan; otherwise $c_{k,k'} = 0$.

Objects

An object is a connected set of components with the same gray level $\alpha = 1$.

Object Labels

Every object is assigned a unique integer called label.

2.3.2 Object Labelling and Segmentation

The object labelling method proposed here is similar to the method discussed in Rosenfeld and Pfaltz [1966]. Starting from the bottom left corner, every image is scanned column-by-column from bottom to top. Information regarding the starting and ending coordinates of every component and object label encountered at the i^{th} scan is recorded in a tuple ($T_i^k(j)$, j=1, 2, 3, 4) where k denotes the component number and the last field is used to store the object labels. Two lists,

$$L_i : (T_i^1(j),\ T_i^2(j),\ \ldots, T_i^k(j),\ \ldots, T_i^m(j))$$

and

$$L_{i-1} : (T_{i-1}^1(j),\ T_{i-1}^2(j),\ \ldots, T_{i-1}^{k'}(j),\ \ldots, T_{i-1}^n(j))$$

of tuples, formed at the i^{th} and $(i-1)^{th}$ scans are used for the object lebelling. Labels are assigned by examining the component connectivity of lists L_i and L_{i-1}. To establish the connectivity among objects, after each scan a C-matrix is created.

The C-matrix together with the two vectors $A_{m,1}$, where $a_k = \sum_{k'=1}^{n} c_{k,k'}$ and $B_{1,n}$, where $b_{k'} = \sum_{k=1}^{m} c_{k,k'}$ are used to detect the starting, termination, continuation, split and merging states of regions. A C-matrix and vectors $A_{m,i}$ and $B_{i,n}$ at the i^{th} and $(i-1)^{th}$ scans of Fig. 2.3(a) are shown in Fig. 2.3(b).

The component coordinates belonging to the different objects are recorded in separate lists. These lists are used for the computation of the area, span etc. In subsequent discussion these lists are denoted by P_s where s is the object label. Conditions to detect the various states of an object and the labelling process involved under different states of objects are discussed next.

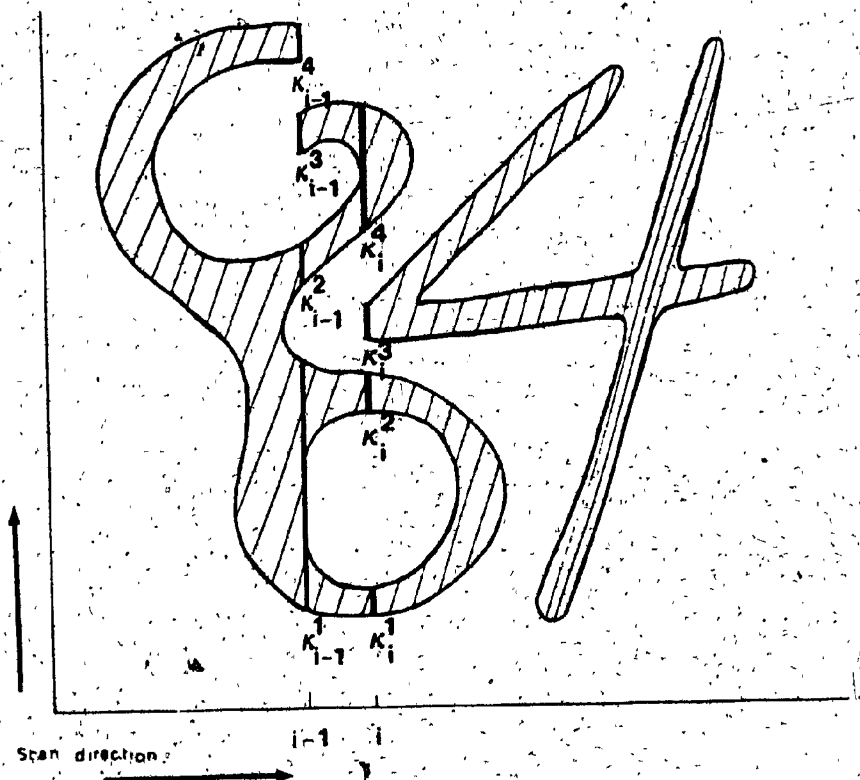


Fig. 2.3 (a). Scanning Method and Component Definition.

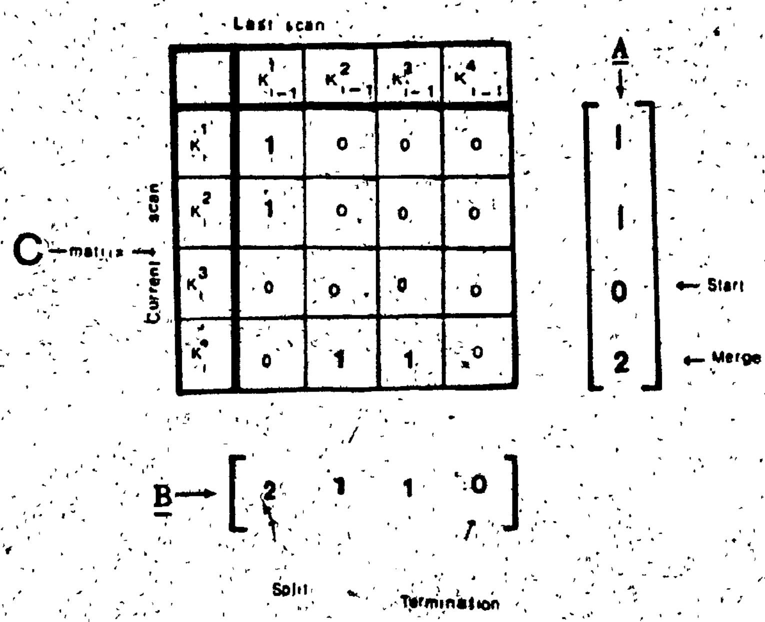


Fig. 2.3 (b). C-matrix and State Detection.

Start of New Objects

The value of $a_k = 0$ where $a_k \in A$ and $1 \leq k \leq m$, implies the beginning of a new object of the k^{th} component encountered at the i^{th} scan. A new label is stored in $T_i^{j}(4)$. Starting and ending coordinates of this component are recorded in list $P_{S=T_i^j(4)}$.

Termination of Old Objects

The value of $b_{k'} = 0$ where $b_{k'} \in B$ and $1 \leq k' \leq n$, implies the termination of an existing or a limb of a continuing object.

Continuation of Old objects

The values $a_k = b_{k'} = 1$ and $c_{k,k'} \neq 0$ where $a_k \in A$ and $b_{k'} \in B$ imply the continuation of the k'^{th} component from the $(i-1)^{th}$ scan to the k^{th} component at the i^{th} scan. In this case, the label $T_{i-1}^{k'}(4)$ associated with the k'^{th} component is assigned to $T_i^k(4)$. The starting and ending cordinates of the k^{th} component is appended to the list $P_{S=T_{i-1}^{k'}(4)}$

Split of Objects

The value $b_{k'} > 1$ where $b_{k'} \in B$ implies the split of objects from the k'^{th} component of the $(i-1)^{th}$ scan. Each non-zero element in the k' column of the C-matrix identifies an splitting object. Each splitting component is assigned the label of the source component, that is, for every k, $1 \leq k \leq m$, if $c_{k,k'} \neq 0$ then label of the source component $T_{i-1}^{k'}(4)$ is assigned to $T_i^k(4)$ and the starting and the ending coordinates of every splitting component k are appended to the list $P_{S=T_{i-1}^k(4)}$.

Merge of Objects

The value $a_k > 1$ where $a_k \in A$ implies the merge of objects from the $(i-1)^{th}$ scan to the k^{th} component at the i^{th} scan. Each non-zero element in the k^{th} row of the C-matrix identifies a merging object. In this case, the labels are assigned according to the following rules.

Rule 1

Find the label of the oldest object among the merging objects. This can be determined by taking the minimum value of all the labels of the merging region, thus the oldest label $\eta = \min [T_{i-1}^{k'}(4)]$ for all $k'=1, 2, \ldots, m$ for which $c_{k,k'} \neq 0$.

Rule 2

Assign η to $T_i^k(4)$ as the label for the k^{th} component at the i^{th} scan and append the starting and ending coordinates of this component in the list P_η.

Rule 3

Modify the labels of the i^{th} scan components, that is, for all j, $1 \le j \le m$ and $j \ne k$, if $T_i^j(4)$ contains the same label as one of the merging components then assign η to $T_i^j(4)$.

Rule 4

Modify the labels of the merging components, that is, for all k', $1 \le k' \le n$, if $c_{k,k'} \ne 0$ then append list $P_{S=T_{i-1}^{k'}(4)}$ to the list P_η and assign η to $T_{i-1}^{k'}(4)$.

The labelling procedure for a single scan described above is repeated during each scan. The final outcome of the entire process is the number of object lists which corresponds to the total number of objects in a ZIP code image. The area (total number of dark pixels) and span (horizontal and vertical) of each object can be computed from these lists.

If the total number of seen objects is the same as the expected number of objects ("five") then the image is assumed to be of good quality and considered for further processing; otherwise, the following processes are performed:

Case I. Number of Seen Objects is Less Than Expected

If the total number of seen and expected objects differs by one or two then it is assumed that some objects are touching each other. The touching objects are identified with the help of horizontal span, and segmentation is performed by the "projection-based" method. The performance of this operation was found to be poor.

Case II. Number of Seen Objects Greater Than Expected

This may happen due to--break in the individual object, "salt-pepper" noise, field intrusion or extraneous object etc. In this case all objects having areas less than a threshold (10% of the largest object) are ignored. If the remaining objects are still greater than five, then all the objects, spanning along the top and bottom borders and any other objects not spanning vertically through the medial line are discarded. If the number of remaining objects is still greater than five then the image is submitted to the binarization process; otherwise, it is accepted. It should be noted that for any image, if the process of binarization and segmentation is repeated for more than the specified number of times (three times) then that image is discarded. An example of a discarded image is shown in Fig. 2.4 and some samples of the correctly segmented ZIP codes are shown in Fig. 2.5. Due to the poor quality of the ZIP code images, approximately 40% of the ZIP codes were discarded. From these discarded ZIP codes some useful character images were manually extracted. 98% of the processable ZIP codes were correctly segmented. The major sources of segmentation errors were the acceptance of some extraneous objects (field intrusions, ink spatters etc.) as character images and failure in identifying the touching characters.

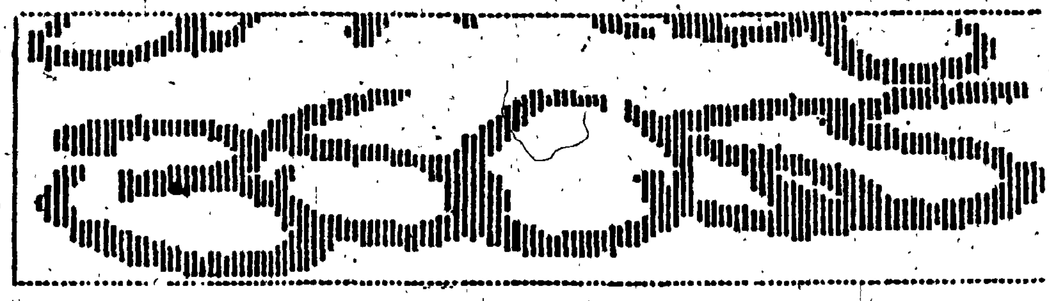


Fig. 2.4. A Typical Rejected ZIP code.

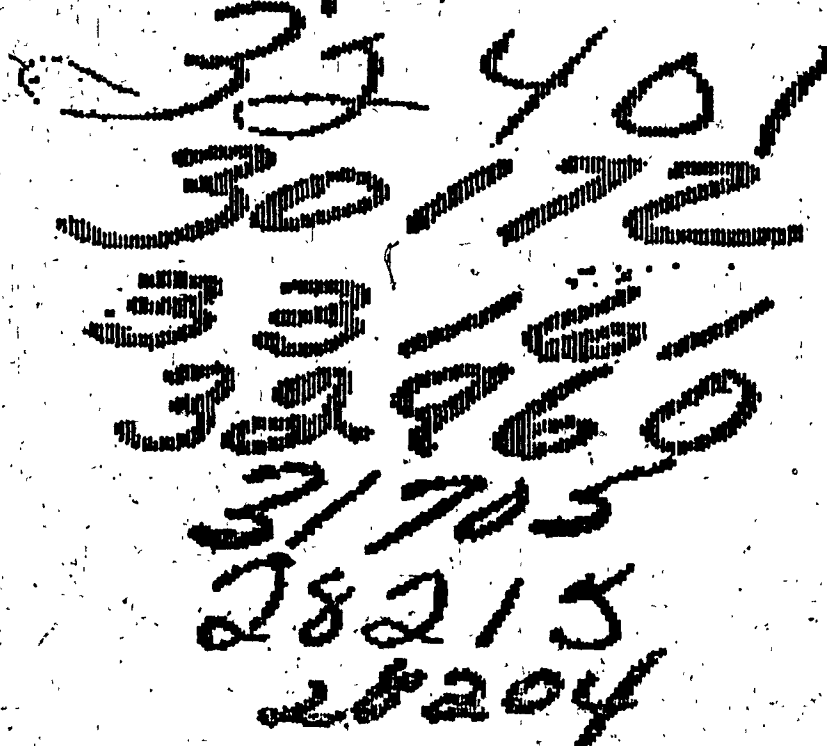


Fig. 2.5. Some Successfully Segmented ZIP code Samples.

Chapter 3

Feature Definition, Extraction and Knowledge Organization

3.1 Introduction

The performance of a recognition system (in terms of speed and reliability) highly depends on the feature quality and feature extraction algorithms. Furthermore, recognition techniques vary widely according to the feature definition, feature selection, the way these features are extracted, and the classification schemes (Suen et al [1980]). For the special application, such as the development of the character recognition system, some criteria have been laid to measure the quality of features. Suen et al [1980] have suggested two basic criteria: (1) sensitivity to the deformation (noise, distortion, style variation, translation and rotation) of a character image and (2) practical implementation of a feature extraction technique, which ofcourse can affect the reliability and speed of a recognition system. These two very general criteria are not only applicable to the character patterns, but also equally applicable to the general pattern recognition application.

During the past several decades considerable research has been done to define and extract the good quality features (Suen [1982a]; Levine [1969]). Generally speaking, features used to solve a pattern recognition problem can be grouped into two wide groups: numerical features and structural features. Numerical features are generally used to deduce the global information while the structural features are used to derive the global structure of a pattern. In character recognition applications the use of numerical features has been studied intensively. Commonly used numerical features are: distribution of black and white points in overlapping or nonoverlapping zones, (Hussain et al [1972]), moments about a chosen point (Spanjersberg [1974]), n-tuples (Bledsoe and Browning [1959]), characteristic loci (Knoll [1969]), multi-directional loci (Suen [1982a]), crossing (transitions from black to white region and vice versa) (Doyle [1960]; Calvert [1970]), distances of points from a reference point (Chen [1965]; Fu et al [1967]); transformations and series expansions (Granlund [1972]; Andrews [1971]; Guedesen [1976]; Wendling and Stamon [1976]); and template matching and correlation coeffecients (Shimura [1973]).

Algorithms to extract the numerical features are easy to implement but these features are highly sensitive to style variations, translation and rotation, and do not provide the structural description of a pattern. The use of structural features reflecting the geometrical and topological properties of a pattern has been suggested to account for the variation in pattern (Suen [1982b]). Commonly used structural features are strokes, bays (cavities), end points, intersections of line segments, and loops (Hunt [1972]; Munson [1968]; Parks et al [1974]; Tou and Gonzalez [1972]; Watt and Beurle [1971]; Ali and Pavlidis [1977a], [1977b]).

Unlike the numerical features, extraction of the structural features is difficult. Design of algorithms to extract the structural features is still a topic of research. A scheme capable of extracting the structural properties from a pattern is developed and implemented. Details of this new scheme are presented in section 3.3.

One of the aims of this research was to study the performance of a combination of the structural and the numerical features on the totally unconstrained handwritten numeric patterns. An extension of the multi-directional characteristic loci features are used as the numerical feature.

3.2 Numerical (Statistical) Features

It has been observed that even in the field data like totally unconstrained handwritten ZIP codes, a large percentage of samples do not vary from their standard shape and sizes. They do possess some degree of consistency. Thus application of purely structural analysis on all the samples is not a good choice. The numerical features could be used for the faster processing of such samples. The method to extract these features is described below.

Starting from the top left corner, every input pattern is scanned row-by-row. At each boundary point a list consisting of the following information is generated, see Fig.3.2.1.

- Boundary point co-ordinates (row number and column number).

- Black region counts in the north, east, south and west directions (C_N, C_E, C_S, and C_W). In each direction the counts are coded--no region as "0", one region as "1", two regions as "2", three regions as "3" and four regions as "4". Counts more than four which seldom occur are set to four for our application.

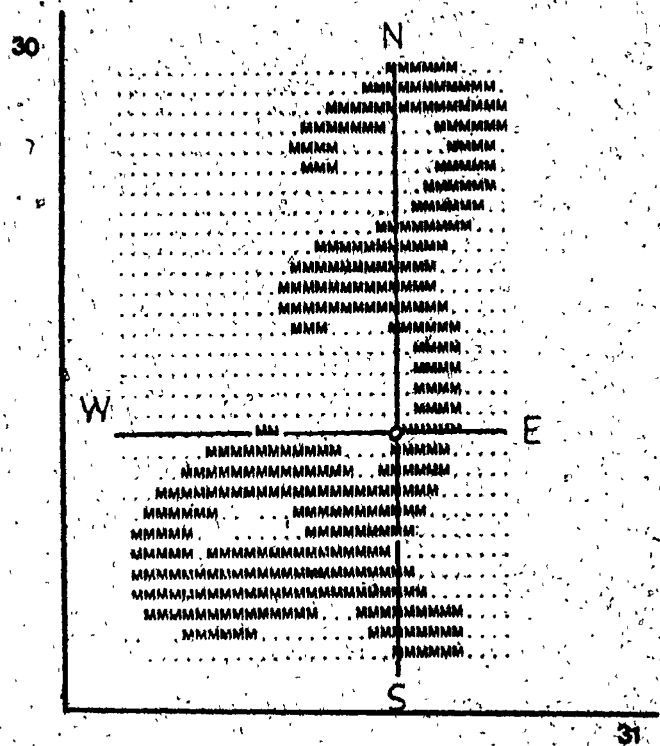
- Adjacent black run length (B_N, B_E, B_S, B_W) in the north, east, south and west directions.

- Adjacent white run length (W_N, W_E, W_S and W_W) in the north, east, south and west directions.

White and black run lengths are quantized on the scale of four--no black or white run length as "0", short as "1", medium as "2", large as "3" and extra large as "4". In the cases of north and south directions, if a run length is less than one-fourth of the height of the pattern then it is coded as short; if it lies between one-fourth and half of the height, it is coded as medium; if it lies between half and three-fourths of the height, it is coded as large; otherwise coded as extra large. Using the width of a pattern, similar codes are assigned to the black and white run lengths for the east and west directions.

Finally, using the boundary coordinates, height and width of the pattern, the zone numbers for boundary points are calculated. Each character image is divided into 20 rows and 15 columns, this partitioning was done by incorporating the average height to width ratio.

All feature values are stored in a table of size M x n, where M is the number of zones and n denotes the number of features. This table is used to decode the features as $u_{ijk}=1$ if the j^{th} feature takes value $k \in (0, 1, 2, 3, 4)$ in the i^{th} zone; otherwise it is assigned a value 0.



Codes

C_N	C_E	C_S	C_W
2	x	1	1

Black Runlength

B_N	B_E	B_S	B_W
x	1	1	x

White Runlength

W_N	W_E	W_S	W_W
1	x	x	2

Fig. 3.2.1. Statistical Feature Codes.
"x" represents Nil.

3.3 Shapes and Shape Features

The objective of the feature definition is to find the characteristic(s) of the patterns which can be used to distinguish one pattern from another, and shape is such a characteristic. The importance of shape features has long been recognized in solving many practical pattern recognition problems. The study of shapes is an active field in pattern recognition (Bribiesca and Guzman [1980]; Mori and Doh [1982]; Danielsson [1978]; Agrawala and Kulkarni [1977]; Davis [1977]; Persoon and Fu [1977]; Zahn and Roskies [1972]; Arcelli and Levialdi [1973], [1971]). In the survey on character recognition application, Suen et al [1980] recommended the syntactic approach as a possible solution to achieve the optimal recognition result. In fact, the success of such an approach (syntactic/structural) highly depends on the precise definition of shape primitives.

Shape is an abstract concept. There is no rigorous definition of shapes. Bribiesca and Guzman [1980] have given the following definition of a shape.

> "A shape is what remains of a region after discarding its size, position and orientation."

The above definition seems to be very close to our intuitive definition of shapes. There have been efforts to realize such an abstract concept, and techniques were developed to describe shapes. Generally speaking these techniques fall into two major categories.

In the first category, which may be termed as the structural approach, shapes are described in terms of the inter-relationships among the basic constituents of a shape (Davis [1977]; Shaw [1969]). For example, a rectangular shape can be defined using line segments and their angular relationships. The basic constituents (primitives) used in this approach vary from application to application. However, Freeman's [1970] directional codes have been widely used as primitives.

In the second category, which may be termed as the functional approach, shapes are described in terms of some algebraic relationships among some quantitative measurements (Zahn and Roskies [1972]; Persoon and Fu [1977]; Underwood [1970];Minsky and Papert [1969]). For example, the quantity P2A:

$$P2A = P^2/4\pi A$$

P = perimeter,

A = area

which can be used as a measure of circularity (Danielsson [1978]) when normalized to unity. These quantities are often referred to as shape factors. A list of such shape factors can be found in Underwood [1970].

3.3.1 Extraction of Shape Features.

Considerable attention has been paid to design good algorithms to extract the shape features. Mori and Doh [1982] have classified these algorithms into three classes: parallel algorithms; Contour tracing algorithms and sequential tracking algorithms, and pointed out that the first two approaches have problems in practical applications and their potentials (speed) can only be exploited by using special purpose hardware. These algorithms are appropriate for simple patterns such as constrained machine printed and handprinted characters, but unsuitable for complex patterns such as unconstrained handwritten characters.

The last approach was originally proposed and used by Rosenfeld and Pfaltz [1966] to extract blobs from a pattern. Later it was used to extract the complex shape features such as concavities, end points, area, perimeter and moments etc. (Shelman [1972]; Agrawala and Kulkarni [1977]; Nadler [1974]; Mori and Doh [1982]).

In the sequential approach, a pattern is either scanned row-by-row (or column-by-column). The information gathered during any two consecutive scans is used to extract the desired characteristics. On the contrary in other approaches a pattern is scanned using a window of a fixed size. Thus, depending on the window size, several tests are required for the pixel to pixel movement. Furthermore, due to the window size limitation, these approaches provide only the local information. These factors make these approaches time consuming and complex. Hence, a simple and fast sequential tracking algorithm developed for this study is described in the next section.

3.4 A New Method to Extract the Shape Features

Methods to extract the shape features entirely depend on the definition of shapes to be extracted. Existing shape extraction algorithms (parallel, contour tracing or sequential) assume the existence of predefined shapes in a pattern. For example, Mori and Doh [1982] have described a method to extract a set of cavities defined in (Fig.3.4.1). But in practical applications due to immense variations in patterns it is extremely difficult to perceive all possible shape variations in advance. Thus, any method based on the predefined shapes may not be flexible enough in capturing the variations in unknown patterns. A pattern recognition system built on this approach can extract only those shapes which are induced during its design phase. It neither can learn shape variations nor can update its knowledge in real-time.

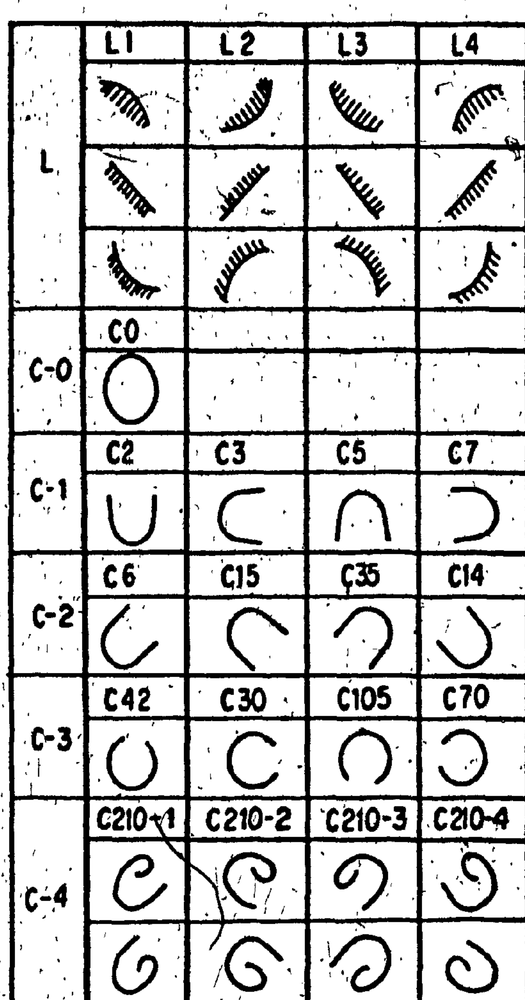


Fig. 3.4.1. A Set of Cavities Defined by Nori and Doh

Most shape extraction algorithms operate in stages. In the first stage local shape features (primitives) are extracted. In subsequent stages, using these primitives or higher level shapes and some shape derivation rules, even higher level shapes are extracted. There are several approaches to implement these algorithms. Among the best approaches is the implementation based on the theory of formal languages. Nadler [1974] described a method to construct a one stage automaton to extract the cavities while Mori and Doh [1982] described a two stage automaton. Though these schemes enjoy the support of the established theory of the formal languages and suffer less conceptual distraction--they are computationally expensive and lack the support of practical results. Furthermore, all possible states of an automaton which can arise due to variations cannot be perceived in advance--it might lead a system into some indeterminate state.

For practical applications a shape extraction algorithm must be simple, deterministic and dynamic in nature. To meet these requirements a method based on edge type classification and the inherent relationship between edges has been investigated and implemented. The approach presented here does not depend on a priori intuitive shape definition, instead, it is exploratory in nature. Depending upon the scanning method it gives the inherent shapes which constitute the fundamental shape. Using the frequency of occurrence of these fundamental shapes in real-life samples, the description of higher level (derived) shapes can be built.

3.4.1 Basic Edge Types

Edges play an important role in analyzing and describing a pattern/subpattern. In a binary pattern an edge is the transition from a white region to a black one or vice-versa, and can be detected while scanning a pattern.

In a horizontal (raster) scan, new edges are detected either by the appearance or a split in a body region. Considering these situations as "context" we can only get any one of the four basic types of edges. These are outer left (e^1), outer right (e^2), inner left (e^3) and inner right (e^4) (Fig.3.4.2).

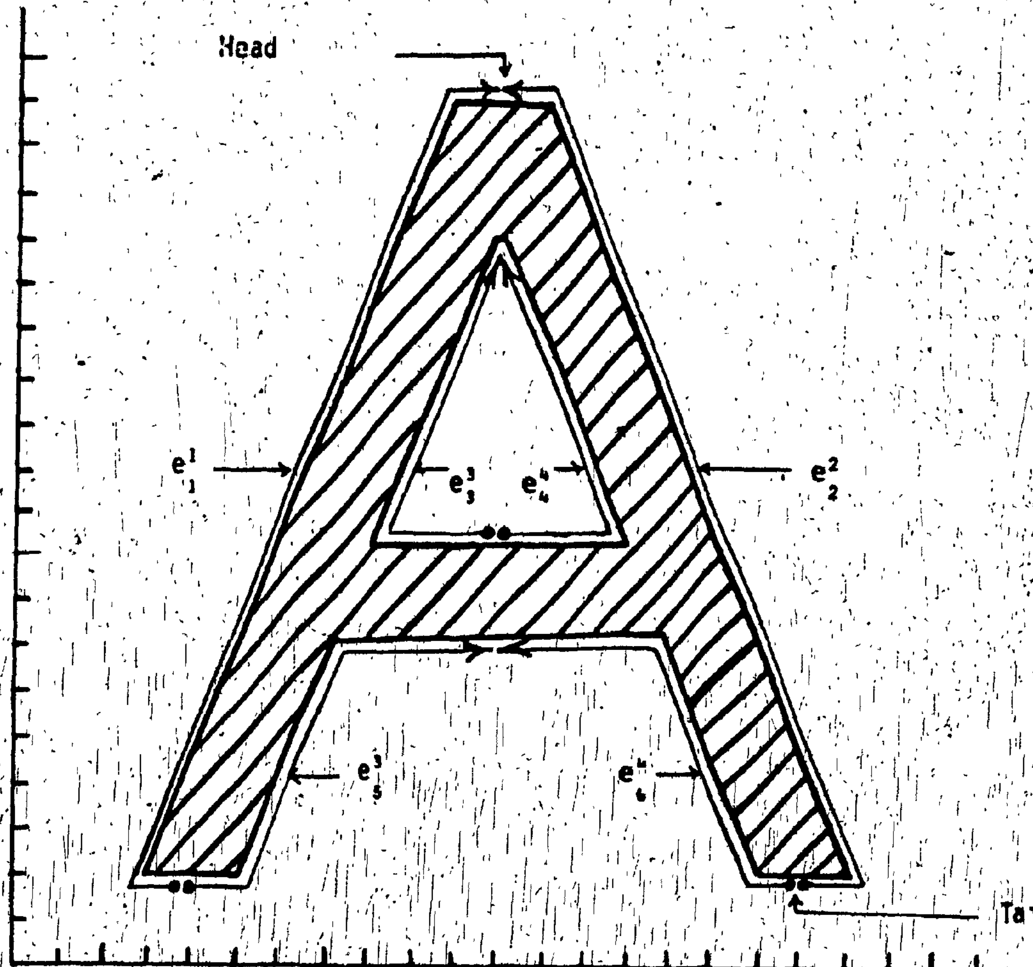


Fig. 3.4.2. Basic Edge Definition.

3.4.2 Edge-to-edge Relationships

An edge begins with the start/split of a body region and terminates with the termination/merge of a body region (Fig.3.4.2). The starting point of a body region is considered as the "head" and the termination point as the "tail" of an edge.

As pointed out in Chapter two that in a sequential scan, a body region can be in one of the four states i.e. start, split, merge or termination state. These states can be easily detected using the conditions given in Chapter two and reported in Ahmed and Suen [1982].

The mutual relationships between different types of edges can be described using some concatenation relations. For simplicity, the natural concatenation relationships, that is, the head of an edge can be connected to only the head of another edge or the tail of one edge can connect only the tail of another edge is considered in this implementation. Using Shaw's [1969] concatenation relations "X" (head-to-head) and "-" (tail-to-tail) and edge types e^1, e^2, e^3 and e^4, all the possible relationships formed by the different edges in a pattern is summarized in the table below where "X" and "-" denote the head-to-head and tail-to-tail concatenations.

	e^1	e^2	e^3	e^4
e^1		X, -	-	
e^2	-			-
e^3	-			X, -
e^4		-	-	

The relationship e_i^k x e_j^l, $i \neq j$, $k \neq l$ represents the head-to-head concatenation of the i^{th} (k^{th} type) edge with the j^{th} (l^{th} type) edge. Similarly, e_i^k - e_j^l means the concatenation of the tail of the i^{th} edge with the tail of the j^{th} edge.

It is noted that the above table summarized the only possible relationships among the edges in any binary pattern.

The relationships between different edges are stored in a table $T=(t_{i,j})$, $(1 \leq i,j \leq n)$, where, n is the total number of edges in a pattern. This table will be referred to as edge table in subsequent sections. Depending upon the relationships among the k^{th} and l^{th} edges of different types, a relationship code (R_1, R_2, R_3, ..., R_{12}) is stored in the the edge table; these codes and the corresponding pattern formations are shown in Fig.3.4.3. The edge table obtained from Fig. 3.4.4(a) is shown in Fig.3.4.4(b).

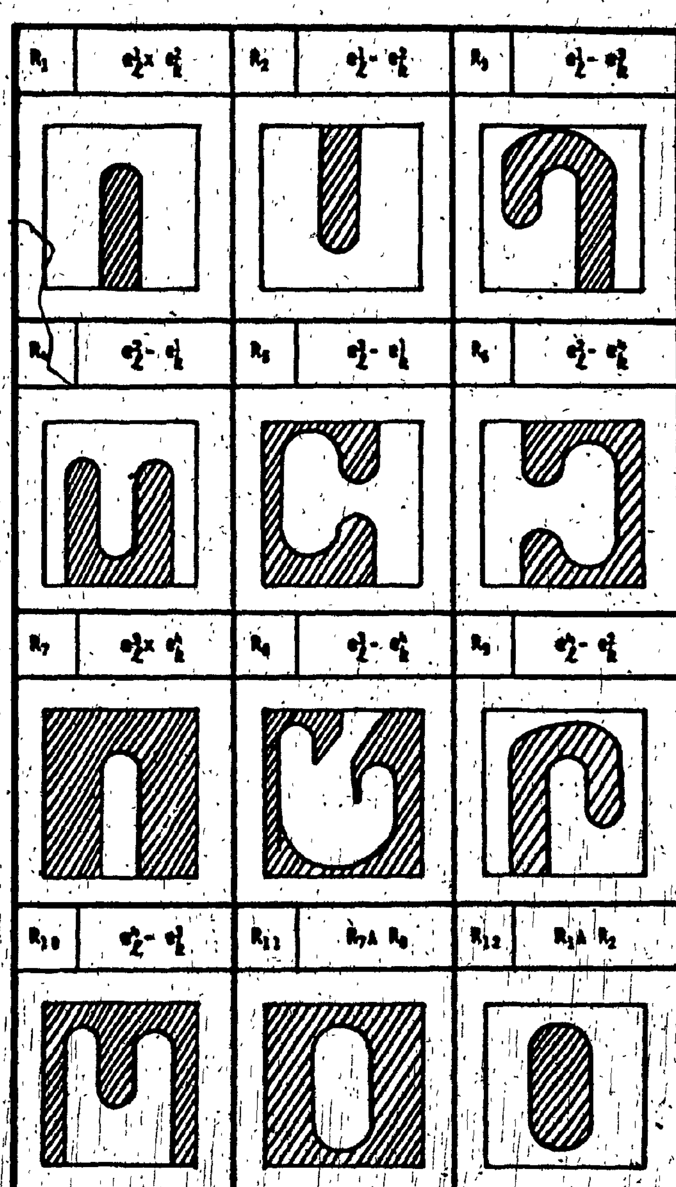


Fig. 3.4.3. Twelve Possible Shapes.

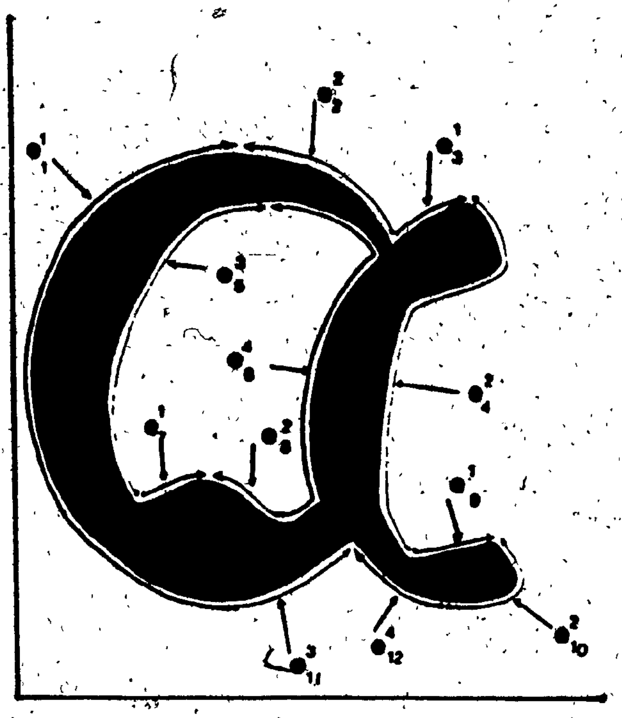


Fig. 3.4.4 a. A Pattern and Its Edges.

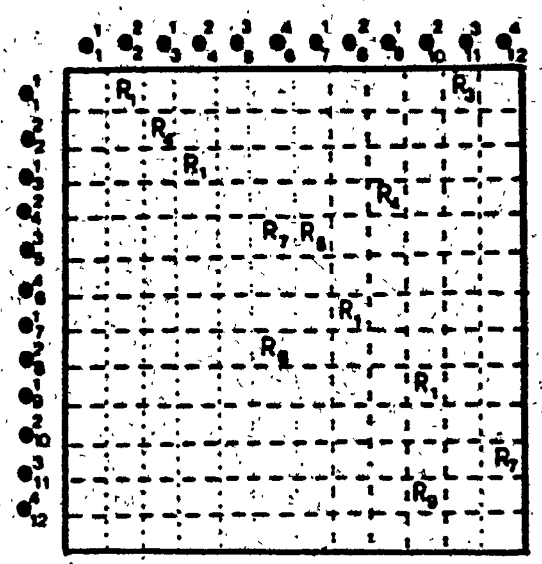


Fig. 3.4.4 b. The Edge Relationship Table.

C^1 : $\{R_7, R_6, R_1, R_5\}$

C^2 : $\{R_7, R_5, R_1, R_4, R_1, R_4, R_1, R_3\}$

h^1 : $\{(e^3_5, e^4_6), (e^4_6, e^2_8), (e^1_7, e^2_8), (e^3_5, e^1_7)\}$

h^2 : $\{(e^3_{11}, e^4_{12}), \;\ldots\ldots\; (e^1_1, e^3_{11})\}$

Fig. 3.4.4 c. Chain Definitions.

3.4.3 Shape Detection

In a pattern, shapes can be classified into fundamental shapes and derived shapes. Fundamental shapes are those characteristics obtained during the scanning of a pattern by a specific method. Line elements of one unit length (Freeman's directional codes) are generally used as fundamental shapes (primitives) in contour tracing or parallel schemes. In the sequential tracking schemes, patterns formed by dark elements in the two consecutive scans can be used as fundamental shape elements. Mori and Doh [1982] have used a complete set of ten fundamental shape elements to derive more complex shapes. These shapes are all the possible combinations of horizontal lines (components) in two consecutive scans. They called these shapes quasi-local features.

Derived shapes are those obtained by a combination of different fundamental shape elements. For example, shapes like line segments which have more than one unit in length, complex cavities of various shapes and orientations, and holes of different shapes and sizes can be obtained by considering the relationships among the fundamental shapes.

In this study, cavities and end points facing different directions (up, down, left or right etc.) and hole are used as shape features (Fig.3.4.5). Simple forms of these features can be easily deduced from the edge type classification and edge to edge relationships. The approach presented here does not require much processing. Instead, simple shape features can be extracted in one scan while scanning a pattern in a desired direction (horizontal, vertical, diagonals). The application of edge types and edge relationships for the extraction of fundamental as well as some derived shapes during the horizontal scan is described in the sections to follow. A similar method is used to obtain the shapes while scanning the pattern in different directions.

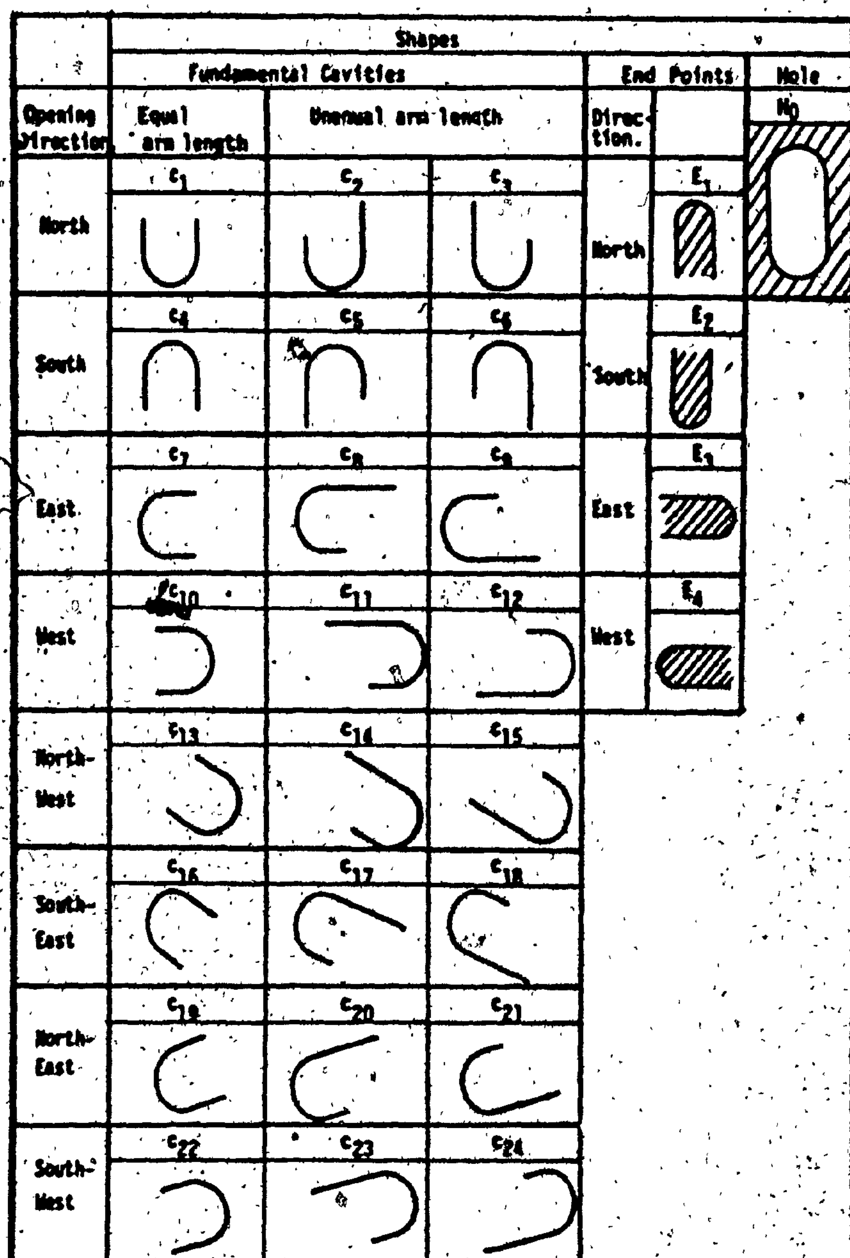


Fig. 3.4.6. Some Shapes Used As Shape Features.

3.4.3.1 Fundamental Shape Detection

Let e_i^k and e_j^l be the i^{th} and j^{th} edges of k^{th} and l^{th} types respectively involved in the formation of a shape. Conditions to detect different fundamental shapes are:

(a) The Beginning of a Blob

The relationship $e_i^1 \times e_j^2$ where $i \neq j$ implies the beginning of a blob formed by the i^{th} and j^{th} edges.

(b) The End of a Blob

The relationships $e_i^1 - e_j^2$, $e_i^1 - e_j^3$, $e_i^4 - e_j^2$ or $e_i^4 - e_j^3$ imply the end of a blob involving the i^{th} and j^{th} edges.

(c) Fundamental Cavity Type^1

The relationships $e_i^2 - e_j^1$, $e_i^2 - e_j^4$, $e_i^3 - e_j^1$ or $e_i^3 - e_j^4$ imply the formation of a cavity opened at the top by the i^{th} and j^{th} edges.

(d) Fundamental Cavity Type^2

The relationship $e_i^3 \times e_j^4$ implies the formation of a cavity opened at the bottom by the i^{th} and j^{th} edges.

The end points are detected from the end or start of a blob. If the limb width is less than 1/5th of the width of a pattern and limb length greater than the 1/3rd of the height of a pattern then a blob is considered as an end point. The positions of the end points are captured by dividing the pattern matrix into 16 (4 rows and 4 columns) zones.

3.4.3.2 Rank Based Cavity Classification

The rank of an edge is a unique positive integer. Every edge is assigned a rank in increasing order according to the order of their appearance. The fundamental cavities can be further characterized by the ranks of the edges involved in its formation. Suppose that two edges e_i^2 and e_j^1 with respective ranks $r(e_i^2)$ and $r(e_j^1)$ are involved in the formation of the cavity of type[1], then the three different cavities formed are: $r(e_i^2) < r(e_j^1)$, (left branch is bigger, or both the branches are equal) and $r(e_i^2) > r(e_j^1)$ (right branch is bigger).

3.4.3.3 Higher Order Shape Derivation

In this section the use of an edge table for the extraction of a hole and higher order cavity features is described. It is evident that edges form a closed chain in and around a body region. These chains can be easily obtained from the information stored in the edge table. Two lists, one showing the chain edge pairs (h^1 and h^2) and other describing the mutual relationships among edges (C^1 and C^2) obtained from Fig. 3.4.4(a) are shown in Fig.3.4.4(c).

3.4.3.3.1 Detection of a Hole Feature

Using the chains of edges, two kinds of holes can be identified: a simple hole which can be detected directly from the relationship R_{11}. Another type of hole, a complex hole which can be detected by analyzing the chains of edges, see Fig.3.4.6(a). It should be noted that a chain encircling a blob and the one forming a hole differs in only one respect. A chain which forms a hole must contain at least one edge pair with the relationship R_7 whose rank is smaller than all other pairs with R_2 relationships. Referring to Fig.3.4.4(a) and Fig.3.4.4(c), the ranks of the edge pair (e^3_{11}, e^4_{12}) with relation R_7 is greater than the ranks of all edge pairs ((e^1_1, e^2_2), (e^1_3, e^2_4) and (e^1_9, e^2_{10})) with relation R_2. Thus, the chain h^1 encircles a blob. On the other hand the ranks of the edge pair (e^3_5, e^4_6) in h^2 is smaller than the ranks of the edge pair (e^1_7, e^2_8) which means the chain h^2 forms a hole. Clearly, this simple test can detect the existence of a hole in any incoming pattern.

3.4.3.3.2 Detection of Higher Order Cavities

Generally speaking, cavities can be classified by the directions of their openings. Various shapes along with openings have also been used for cavitiy classification (Fig.3.4.1). The selection of cavity shapes is arbitrary and it is based on intuition. An intuitive selection may not necessarily reflect the real situation. For example, in the case of the recognition of totally unconstrained handwritten characters it is difficult to predict intuitively the varying shapes of cavities. Furthermore, sometimes it is even difficult to devise methods to extract the defined shape features.

Cavities should be classified on the basis of scan directions and mutual relationship among the different types of edges. Complex cavities can be derived by considering the concatenation relationships of edges with each of the fundamental cavities(type^1 and type^2). Cavities obtained by considering the direct concatenation of edges with the fundamental cavities are referred to as first-order cavities. More complex cavities, such as second and third order and so on, can be obtained by considering the relationships among the edges of first-order cavities and continuing edges in a pattern. The potential of this approach in extracting the first-order cavities is demonstrated in the following example. In our application only the fundamental cavities are used. This section simply highlights the power of this approach for the extraction of complex structural properties from a given pattern.

Suppose that e_i^3 and e_j^4 are two edges of ranks $r(e_i^3)$ and $r(e_j^4)$ with $t_{i,j} = R_7$. The immediate relationships between e_i^3 and some edge e_k^l of rank $r(e_k^l)$ and between e_j^4 and some edge $e_{k'}^l$ of rank $r(e_{k'}^l)$, where $(l=1,2,3,4)$ form the class of cavities. The possible first-order cavities formed by the edge to edge transition are listed below.

e_i^3:

- $R_3 — e_k^1 \;:\; r(e_i^3) > r(e_k^1)$
- $R_3 — e_k^1 \;:\; r(e_i^3) < r(e_k^1)$
- $R_5 — e_k^1 \;:\; r(e_i^3) > r(e_k^1)$
- $R_5 — e_k^1 \;:\; r(e_i^3) < r(e_k^1)$
- $R_8 — e_k^4 \;:\; r(e_i^3) < r(e_k^4)$
- $R_{10} — e_k^4 \;:\; r(e_i^3) > r(e_k^4)$
- $R_{10} — e_k^4 \;:\; r(e_i^3) < r(e_k^4)$
- $R_8 — e_k^4 \;:\; r(e_i^3) > r(e_k^4)$

e_j^4:

- $R_6 — e_{k'}^2 \;:\; r(e_j^4) < r(e_{k'}^2)$
- $R_6 — e_{k'}^2 \;:\; r(e_j^4) > r(e_{k'}^2)$
- $R_8 — e_{k'}^3 \;:\; r(e_j^4) < r(e_{k'}^3)$
- $R_8 — e_{k'}^3 \;:\; r(e_j^4) > r(e_{k'}^3)$
- $R_9 — e_{k'}^2 \;:\; r(e_j^4) < r(e_{k'}^2)$
- $R_{10} — e_{k'}^3 \;:\; r(e_j^4) < r(e_{k'}^3)$
- $R_{10} — e_{k'}^3 \;:\; r(e_j^4) > r(e_{k'}^3)$
- $R_9 — e_{k'}^2 \;:\; r(e_j^4) > r(e_{k'}^2)$

All of the above transitions can be detected from the edge table. Any cavity feature present in a pattern can be obtained by detecting a R_7 relationship in the edge table. By considering combinations of the above transitions, 64 different first-order cavity features can be detected. Some first-order cavities formed by the transitions from e_i^3 to e_k^1 and e_j^4 to e_k^2 are shown in Fig.3.4.6(b).

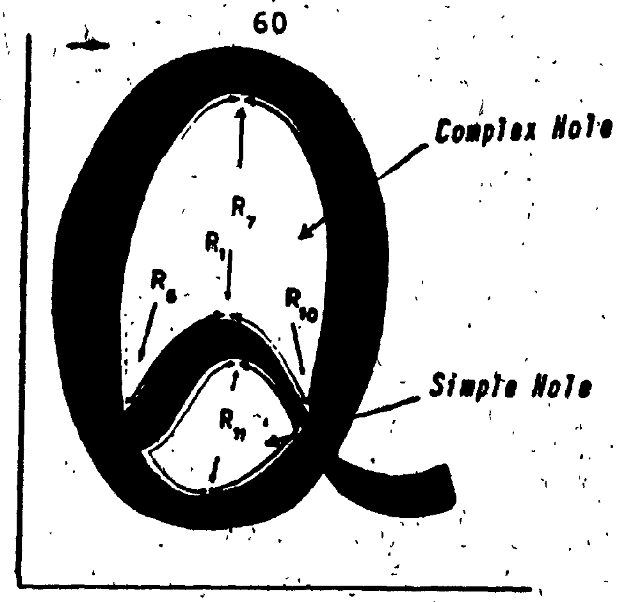
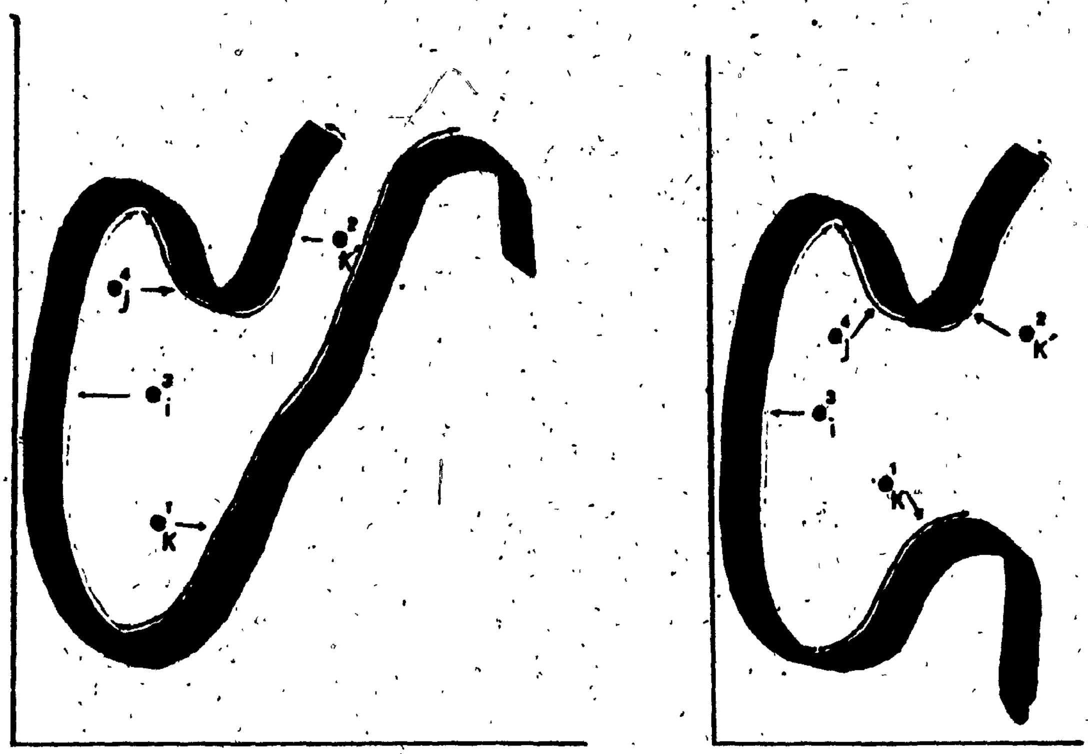


Fig. 3.4.6 a. Hole Types.



$e^3_i \xrightarrow{R_5} e^1_k : r(e^3_i) < r(e^1_k)$

$e^4_j \xrightarrow{R_5} e^2_{k'} : r(e^4_j) > r(e^2_{k'})$

$e^3_i \xrightarrow{R_5} e^1_k : r(e^3_i) > r(e^1_k)$

$e^4_j \xrightarrow{R_5} e^2_{k'} : r(e^4_j) > r(e^2_{k'})$

Fig. 3.4.6 b. First Order Cavities.

3.5 Knowledge Organization

The information obtained from the training set samples or from some other sources is considered as knowledge and used for future references. It is customary to store a priori knowledge in an explicit or implicit form. Examples of explicit form of storage methods are templates, weight vectors and feature vectors etc. Decision trees and production rules are some examples of the implicit form of knowledge storing mechanisms. In order to increase the efficiency of a system, a knowledge storage and retrieval mechanism should be flexible enough to upgrade the system's knowledge in real-time.

For the processing of pictorial data the usage of the relational data model defined below (Codd [1970]), is proposed to store and organize the knowledge regarding various picture constituents (Kunii et al [1974]; Chang [1981]). In this study, the relational data model is used to store the spatial distribution of shape features and its relevance towards a class of pattern.

3.5.1 Relational Databases

Definition 3.5.1

Given a collection of sets $D_1, D_2, \ldots, D_n$ (not necessarily distinct nor disjoint), R is a relation on these n sets if it is a set of ordered n-tuples $<d_1, d_2, \ldots, d_n>$ such that d_1 belongs to D_1, d_2 belongs to D_2, ..., d_n belongs to D_n. Sets D_1, D_2, ..., D_n are the domains of R. The value n is called the degree of R, and the number of tuples in R its cardinality.

3.5.2 Spatial Distribution of Shape Features

In certain situations, mere occurrence of a feature does not provide sufficient discriminatory information. The physical location of a feature is also required. For example, the occurrence of a hole feature is not sufficient to discriminate between digits 6 and 9. The physical location of the hole is important.

In the conventional approaches, the spatial distributions of features are captured by dividing the two dimensional pattern into several zones. The rigid zone definition limits the study of pattern variations. Furthermore, often zone partitioning is arbitrary and the sizes of the zones are selected without much justification. To overcome this problem a scheme to capture and store the spatial information about each shape in relational databases is implemented. This scheme provides a mechanism to study the cluster formation of shapes in a two dimensional plane. Some shapes used as features in this study are shown in Fig.3.4.5.

In this scheme every shape feature is considered as a relation (Fig.3.5.1(b)). The degree of each tuple is 14. The first four attributes are used to store the normalized coordinates of the bottom left and the top right corners of the best-fit rectangle (minimum rectangle which encloses the entire shape) of a shape feature (Fig.3.5.1(a)). A relevance measure (frequency of occurrence) of a shape feature towards a pattern class (P_1, P_2, ..., P_N) is stored in the remaining consecutive attributes.

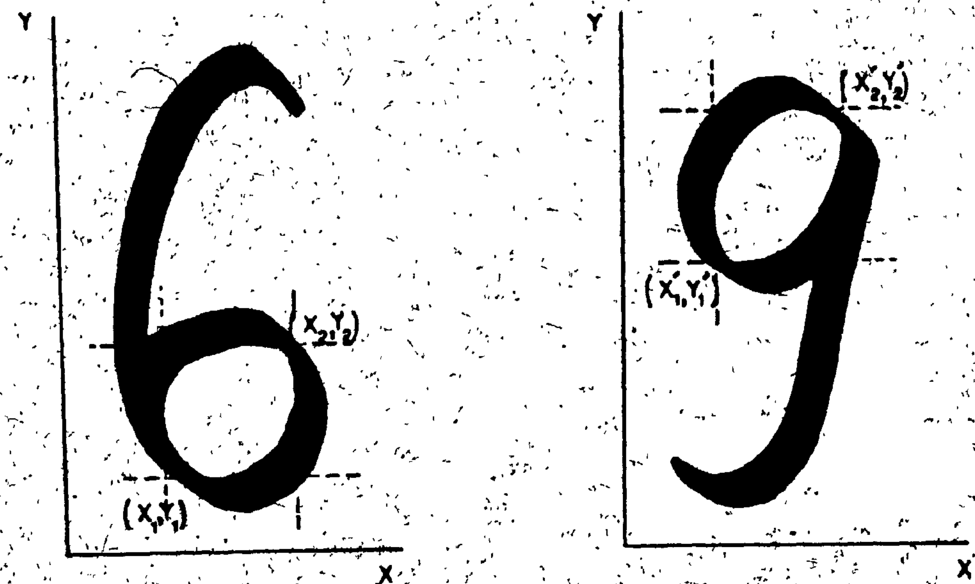


Fig. 3.5.1 a. Spatial Distribution of Shape Feature.

xbottom	ybottom	xtop	ytop	P_1	P_2	...	P_N
x_1	y_1	x_2	y_2	$n_{1,1}$	$n_{1,2}$		$n_{1,N}$
x_1'	y_1'	x_2'	y_2'	$n_{2,1}$	$n_{2,2}$	...	$n_{2,N}$
⋮	⋮	⋮	⋮	⋮	⋮	...	⋮

Fig. 3.5.1 b. Relation Hole.

The first four attributes are used as a key to search for a shape feature in a specified position. This search is accomplished by computing the distances between the bottom-left and top-right corner points of the best-fit rectangles of an incoming shape and stored shapes. Suppose that the coordinates of the bottom-left and top-right corner points of the best-fit rectangles of the incoming shape and stored shapes are $((x_1, y_1), (x_2, y_2))$ and $((x_1', y_1'), (x_2', y_2'))$ respectively as illustrated in Fig. 3.5.1(a). The distances between the two bottom points d_b and the two top points d_t are computed as follows:

$$d_b = \sqrt{(x_1 - x_1')^2 + (y_1 - y_1')^2}$$
$$d_t = \sqrt{(x_2 - x_2')^2 + (y_2 - y_2')^2}$$

If distances d_b and d_t are less than some prespecified thresholds (bottom threshold and top threshold) then the two shapes occupy the same spatial position. In this situation the relevance of the shape in that position towards a pattern class is updated in the database. If the distances are greater than the threshold then a new tuple is created.

It is interesting to notice that the cardinality of each relation is a function of the thresholds. A smaller value of the threshold will result in a larger number of tuples in a relation with lesser ambiguity and a larger value of the threshold will result in a smaller number of tuples with greater ambiguity among the patterns. In order to keep the optimum number of tuples in a relation, the threshold values are decided by allowing a specified level of ambiguity. For different threshold values ranging from 1 up to the maximum size of the pattern matrix, the ambiguities are computed by:

$$E = 1/MxN \sum_{i=1}^{M} \sum_{j=1}^{N} -p_{ij} \log p_{ij}$$

where M is the cardinality of the relation, N is the number of expected patterns and p_{ij} is the estimate of probabilities that a pattern belongs to the j^{th} expected class due to the shape feature inside the i^{th} best-fit rectangle. For our application a threshold of unit radius (shapes occupy the same locations in the two dimensional plane if and only if both distances d_b and d_t are less than one) was found suitable to control the number of tuples in each relation.

Chapter 4

Decision Making Process in Pattern Recognition

4.1 Introduction

The decision making process (classification) in pattern recognition can be formally stated as follows:

Let $X=(x_1, x_2, x_3, \ldots, x_n)$ be a vector of n characteristic measurements (feature vector) obtained from an unknown pattern X, and let $C=(c_1, c_2, c_3, \ldots, c_m)$ be the set of m classes. Based on the characteristic measurement values X, a decision making process either assigns an unknown pattern to one of the known classes $(c_1, c_2, c_3, \ldots, c_m)$ or it rejects the unknown pattern.

Development of a good (minimum error classification) decision making method has been one of the major topics of research in pattern recognition. Methods from various theoretical and applied fields, such as mathematics, statistics, theory of formal languages, graph theory and heuristic etc., have been explored and applied for the purpose of pattern classification. In this chapter, different classification methods are reviewed and categorized on the basis of their underlying operating principles. Although, these methods are applicable to any general types of patterns, the aim of this survey are: (1) to highlight their applicability to the character recognition problem, where all possible variations in handwritings cannot be perceived in advance; and (2) to justify the need for a classification scheme which should update the knowledge-base (real-time learning) without interfering the preprogrammed decision making process.

4.2 Classification Methodologies

The selection of a classification method depends on the domain of an application and the types (numerical/structural) of features extracted from the underlying patterns. For example, if the extracted features represent the structural properties, and the application under consideration requires both the description and classification, then the syntactic classification scheme, discussed in the forthcoming sections, seems to be a reasonable choice. Many different classification approaches have been reported in the pattern recognition literature. These approaches may be loosely placed in the decision-theoretic, syntactic and graph-theoretic categories.

4.2.1 Decision-theoretic Approaches

During the past several decades, development of mathematical models for pattern classification has been a major subject of research. Several deterministic and probabilistic methods were developed and their performances were tested on typewritten and/or handwritten characters. These methods may be distinguished from each other on the basis of the underlying assumptions.

The deterministic methods are based on the assumption that the characteristic measurement values are deterministic. In the probabilistic methods the characteristic measurement values are considered as random variables, and it is further assumed that there exists a multivariate distribution function of the characteristic measurement vector for each class (Nilsson [1965]).

In the deterministic methods the pattern classification problem is formulated in terms of the "discriminant functions" while the probabilistic methods require the estimation of some finite set of parameters to approximate the functional form of the assumed distribution function. The basic concepts involved in the development of the many different forms of discriminant functions and the salient features of the probabilistic methods are briefly presented here.

4.2.1.1 Discriminant Functions

Suppose that n characteristic measurements $X=(x_1, x_2, x_3, \ldots, x_n)$ represent a point in the n-dimensional characteristic space. Then the discriminant function $D_j(X)$ associated with the pattern class c_j, $j = 1, 2, 3, \ldots, m$, partitions the characteristic space into m mutually exclusive regions, where each region corresponds to a particular pattern class--this may happen only when classes are distinctly apart. The classification procedure using the discriminant function is as follows.

Given X from an unknown pattern and the discriminant function D_j (maximum m discriminant functions), the classification of an unknown pattern is to assign the unknown to the class c_i if $D_i(X) > D_j(X)$ for all $i,j = 1, 2, 3, \ldots, m$ and $i \neq j$.

Several functional forms of discriminant functions have been proposed and used in pattern classification. Commonly used forms are linear, piece-wise linear and polynomial discriminant functions.

A linear discriminant function $D_j(X)$ is a linear combination of $X=(x_1, x_2, x_3, \ldots, x_n)$, that is

$$D_j(X) = \sum_{k=1}^{n} w_{jk} \cdot x_k + w_{j,0}$$

Where $W_j = (w_{j,1}, w_{j,2}, w_{j,3}, \ldots, w_{j,n})$ is the weight vector obtained from the j^{th} class training set sample patterns and $w_{j,0}$ is a constant. Several (parametric and nonparametric) approaches to estimate these weight vectors are discussed in (Ullmann [1973]). Linear discriminant functions have been used by Chow [1957], Marill and Green [1960], Highleyman [1962], Minneman [1966], Tou and Gonzalez [1972] and several other researchers in character recognition.

Linear discriminant functions are easy to implement but one of their major disadvantages is that this approach assumes that the individual patterns are separated by neat and clean boundaries, which is in general an unrealistic assumption. Experience in pattern recognition applications shows that often classes are not linearly separable. To achieve the maximum separability, use of the piece-wise linear discriminant functions was proposed.

In the piece-wise linear discriminant approach more than one weight vector per class are used. This approach is formally stated below.

Let W_1, W_2, W_3, ..., W_m be the m sets of weight vectors associated with m classes (c_1, c_2, c_3, ..., c_m) respectively, and let weight vectors in the set W_j be denoted as $W_j^{(k)}$ for $k=1, 2, 3, \ldots, u_j$, where u_j is the number of weight vectors in the set W_j. A piecewise linear discriminant function is defined as:

$$D_j(X) = \underset{1 \le k \le u_j}{\mathrm{MAX}} \left(D_j^{(k)} \right)$$

where $D_j^{(k)}$ is the value of the k^{th} discriminant function for the j^{th} class.

An alternative formulation of this approach considers the m sets of weight vectors $W_1, W_2, W_3, \ldots, W_m$ as the set of reference vectors for the m classes $(c_1, c_2, c_3, \ldots, c_m)$ respectively, and the distance $d_j^{(k)}(X, W_j^{(k)})$ between the k^{th} reference vector of the j^{th} class and the unknown pattern X is used to define the discriminant function as follows:

$$D_j(X) = \underset{1 \le k \le u_j}{\mathrm{MIN}} \left(d_j^{(k)} (X, W_j^{(k)}) \right)$$

The nearest neighborhood method discussed in Cover and Hart [1967] is an example of this approach. The application of this technique in character recognition and its suitability in dealing with the problems, such as, good reference vector selection (Ullmann [1974]) and optimum decision making criterion can be found in (Duda and Fossum [1966]; Johnson et al [1966]; Glucksman [1971]; Deighton and Bass [1974]).

Another functional form which is termed as a polynomial (nonlinear) discriminant function was developed to attain maximum separability through the nonlinear discriminant functions, especially when the classes are not linearly separable. The polynomial discriminant function is in general, defined as:

$$D_i(X) = \sum_{j=1} w_{i,j} \phi_j(X)$$

where $\phi_j(X)$ is a function of the characteristic measurement vector X. Several methods based on orthogonal expansion, least square approximation and stochastic approximation, etc., have been suggested (Ullmann [1973]; Specht [1967] and Meisel [1968]) to obtain the functional form of $D_i(X)$.

The approaches presented in this section may perform well for non-overlapping classes, however, in practice, our observations are that classes often overlap. To deal with this problem other methods were investigated. A commonly used and well studied probabilistic method is briefly presented here.

4.2.1.2 Probabilistic Classification Method

In the probabilistic models, each class c_j is associated with a priori probability $P(c_j)$ and conditional density $P(X/c_j)$, for j=1, 2, 3, ...,m. On the basis of a priori information $P(c_j)$ and $P(X/c_j)$ the function of a classifier is to test m statistical hypotheses that an unknown pattern X belongs to the class c_j by defining a decision function $D_j(X)$. This decision function can be obtained by the Bayes theorem using the a priori and conditional probablities, $P(c_j)$ and $P(X/c_j)$. The general form of this decision function $D_j(X)$ is defined as (Nilsson,[1965]):

$$D_j(X) = P(c_j)\ P(X/c_j)$$

The classification procedure known as Bayes optimum decision rule is used to classify an unknown pattern into one of the known pattern classes. Thus, an unknown pattern X is accepted as a member of the class c_j if $D_j(X) > D_i(X)$, for all $i=1, 2, 3, \ldots, m$ and $i \neq j$.

It is important to note that the computation of class densities requires the assumption about the underlying distribution functions. This assumption may not necessarily reflect the actual distribution of the characteristic measurement vectors obtained from the underlying pattern.

4.2.2 Syntactic Approaches

In various pattern recognition applications, along with the classification, the description of an unknown pattern is also required. The syntactic approach was developed to meet this requirement. Considerable theoretical as well as applied studies have been reported in the pattern recognition literature (Narasimhan [1969]; Narasimhan and Reddy [1971]; Kramer et al [1973]; Fu [1974]; Ali and Pavlidis [1977a], [1977b]; Zhang [1980]).

In this approach a pattern is converted into a string of primitives with specified syntactic operations. Strings generated from known pattern classes or by heuristic rules are used to construct the grammar for a class of patterns. In turn, these grammars are used to design a syntactic classifier, generally known as "syntax analyzer", which is a collection of syntax analysis rules. The main function of the syntax analyzer is to classify an unknown pattern into one of the known classes and provide an adequate description (implicitly or explicitly) of the underlying pattern. Similar to the decision theoretic approaches, this approach may be classified into the deterministic and the stochastic techniques. Some of the basic differences and limitations of these techniques are discussed in the following sections.

4.2.2.1 Deterministic Approach

Suppose that, there are $(c_1, c_2, c_3, \ldots, c_m)$ pattern classes, and $G_1, G_2, G_3, \ldots, G_m$ denote m grammars corresponding to each class, so that the string (for a known pattern) generated by grammar G_i represents the class c_i. Let $L(G_i)$ denote the language (a finite set of strings) generated by the grammar G_i from all training set samples belonging to class c_i. The deterministic classification procedure classifies an unkown string x obtained from an unknown pattern, as a member of the class c_i if $x \in L(G_i)$. In other words, the string x is successfully parsed by the grammar G_i.

It should be noted that in the above scheme a string obtained from an unknown pattern may or may not be parsed. Experience shows that due to variability in patterns, sometimes a given string may be parsed by more than one grammar with a consequent uncertainty in decision making. To deal with this uncertainty the construction of a stochastic grammar was proposed.

4.2.2.2 Stochastic approach

In this approach, for every string $x \in L(G_i)$, a probability $p(x)$, $0 \leq p(x) \leq 1$, $\sum_{x \in L(G_i)} p(x) = 1$, is assigned. During the parsing phase of an unknown string x the probability of acceptance $P(x/G_i)$ by the grammar G_i is computed from the applicable production rule. If string x is successfully parsed by the grammar G_i, the final probability $P(x/G_i)$ is used in decision making, while the string x is rejected by a grammar G_i, $P(x/G_i)$ is set to zero.

Once the parsing of the string x is accomplished and the corresponding probabilities $P(x/G_i)$, for all i=1, 2, 3, ..., m are calculated, a decision-theoretic classification rule, such as the maximum likelihood method may be used to assign the unknown string x to the class c_i if $P(x/G_i)=\underset{G_j}{MAX}\ P(x/G_j)$ for all i, j=1, 2, 3, ..., m, ~~and~~ $i \neq j$.

Some applications and theoretical foundations of this method can be found in Fu [1974]. It is evident that the stochastic approach may help in resolving the uncertainties in the decision making process but it fails to capture the attributes related to primitives or subpatterns. In recent years a new approach has emerged to incorporate the information provided by attributes.

4.2.2.3 Attributed Grammar

The combination of syntactical and statistical approaches has been a topic of research because neither the statistical nor the syntactical approaches alone can provide an adequate solution to the problems encountered in practical applications, such as unconstrained character recognition. The usage of attributed grammars was suggested by Tsai and Fu [1980] for pattern recognition applications. This was not a new idea; Knuth [1968] described the use of attributed grammars to assign semantics to context-free languages.

In attributed grammars, in addition to the syntactic rules, the production rules also consist of semantic rules to incorporate the knowledge about attributes associated with each primitive or subpattern. There are two kinds of attributes; inherited attributes (meaning from the context of a phrase) and the synthesized attributes (meaning built up from within the phrase). In pattern recognition the order of appearance of primitives in a specified scan method may be considered as an inherited attribute while the relative position of subpattern in a plane or the description of a subpattern in terms of size and orientation etc. may be considered as synthesized attributes. The attributed grammar approach is more realistic since it allows the syntax analyzer to accept a string of greater variability.

4.2.3 Other Decision Making Strategies

In addition to the techniques described in the last few sections there have been efforts to explore techniques from other areas such as graph theory and heuristic. No clear boundaries exist between the graph-theoretic and heuristic approaches except that the heuristic approaches apply some rules to capture the common sense decision making process while the graph-theoretic approaches are based on well postulated axioms (Katsuragi et al [1969], Watt and Beurle [1971]; Faris and Behrouz [1983]).

4.2.3.1 Graph-theoretic Approaches

Graph theory has been applied to classify patterns in many different ways; these techniques differ from each other on the basis of pattern representation and decision making procedures. Generally speaking patterns are represented as graphs with or without attributes. In a graph representation without attributes, the special points in a pattern, such as end points, junction points etc., are considered as vertices and lines joining these vertices are regarded as the edges of a graph. On the other hand, in a graph representation using attributes (called attributed graphs) rules are prescribed for assigning attributes to each vertex or edge. The decision making process used in this approach may be stated as follows:

Let G_x and G_r be graphs constructed from an unknown pattern and the reference pattern respectively. If G_x is isomorphic to G_r then x is assigned to the r^{th} class. In case G_x and G_r are attributed graphs and if there exists an isomorphism between G_x and G_r such that the attributes of corresponding vertices or edges differ only within a prescribed threshold, then x is still assigned to the r^{th} class.

It is evident that in order to establish the isomorphism between two graphs, vertices/edges need to be compared and this is a time consuming operation. A special form of graph known as the decision tree may be used to minimize the classification time. Decision trees are constructed using characteristics extracted from the training patterns. The characteristics extracted from an unknown pattern X are compared with the characteristics stored at each node of the decision tree. Depending upon the comparison result of a randomly selected characteristic of an unknown pattern at the root node, a path to an expected subclass is selected. Once the expected subclass is established, the characteristic relevant to that subclass is tested and further decision is made about the unknown pattern. The possible decisions at each node include the correct classification of an unknown pattern, return to root node or further processing at a lower node. This process continues until an unknown is either classified or rejected as an invalid pattern.

This decision making process involves the search of the known pattern. Thus, the design of a decision tree can affect the classification speed. The optimal design of decision trees has been an important topic of research and several methods have been proposed (William and Demetrois [1973], Swain and Hauska [1977], Dattatreya and Sarma

[1980], Lin and Fu [1980]; Wang and Suen [1984]).

4.3 Trends in Decision Making Process

In general, the basic function of a decision making process is application dependent. However, it is expected that every pattern recognition system should (i) incorporate the knowledge provided by the statistical (numerical) and structural characteristic measurements, and (ii) be flexible in learning the variations in patterns. A solution to the former might help in developing a system which can perform satisfactorily in real life applications. Combinations of the structural and the statistical approaches have been suggested by several researchers (Kanal and Chandrasekran [1972], Blackwell [1974], Tsai and Fu [1980]; Duerr et al [1980]) as a possible solution.

The latter is generally addressed by considering a large training set of sample patterns; it is assumed that the training sample patterns reflect all possible variations. A priori knowledge obtained from these samples is used in the form of weight vectors, probability distribution parameters, grammars, prototypes, graphs or decision rules (used in decision trees) etc. A formal knowledge representation method which allows update and organization of knowledge in real-time, without any change in the decision making process, is required. It should be noted, decision making approaches based on syntax analysis or decision trees need re-organization of decision making process for every pattern not perceived during the training session of the classifier. Thus, a formal method for knowledge representation and organization needs to be developed. Based on the theory of relational databases, a knowledge organization and representation strategy, suitable for our classification method is described in the last Chapter and used for the classification in the structral module described in the next Chapter.

Chapter 5

A Multi-stage Classification Scheme

5.1 Introduction

It has been pointed out in the last Chapter that the statistical and the structural approaches alone cannot provide adequate solutions to the problems faced by a recognition system in practical situations, such as in the recognition of totally unconstrained handwritten characters. The statistical approaches are weak in dealing with the problems of shape and size variations while the structural approaches are very sensitive to the random noise and other deformations like unexpected breaks, holes, and limbs in a pattern. A combination of the statistical and structural approaches may be useful in dealing with these problems. There are many different ways in which these approaches can be combined.

In this research, the classification process is divided into two phases: using the numerical features in the prediction module, and the structural features in the structural module Fig.5.1. Performances of individual module and a combination of the two modules for the correct classification are investigated.

$$\boxed{\text{Prediction Module}} \xrightarrow[\text{Expected Classes}]{} \boxed{\text{Structural Module}}$$

Fig. 5.1

The statistical features were used in the prediction module with the primary objective of predicting the expected class of an unknown pattern. But, if the prediction towards a particular class is very strong then the unknown is recognized within this module; otherwise it is passed to the structural module along with the information regarding the expected classes.

The structural features were used in the structural module. In this module if a pattern is recognized as one member of the expected classes then it is accepted otherwise rejected.

In both modules concepts from the fuzzy set theory were used for the classification of an unknown pattern. Concepts and basic definintions used are presented below. The detailed descriptions of each module are described in the subsequent sections.

5.2 Fuzzy set and Pattern Recognition

The fuzzy set theory due to Zadeh [1965], has been applied to develop a practical pattern recognition system. Although the applicability of this theory against the Bayesian approach has been questioned on some philosophical and computational grounds (Stallings [1977]), yet researchers from the applied pattern recognition fields have argued and justified its usage on the grounds outlined below (Kickert and Koppellaar [1976]):

> "... The very high variability in handwritings was the reason for considering the new proposal for the actual class assignment procedure based on fuzzy set concepts. In this method the rather dubious assumption about underlying probability density functions of handwritings is avoided. Moreover, the concept of vagueness seems to be a more appealing and convincing way of describing the variability in letters than the concept of probability."

The fuzzy set theory is used in various ways to solve different problems in pattern recognition and image processing. In syntactical pattern recognition it is used to compute the membership grade of a string of primitives generated from an unknown pattern (Wee and Fu [1969], Depalma and Yau [1975]; Thomason [1975]). Tamura et al [1971] have introduced the notion of fuzzy relations to compute the class-to-class similarity to determine the pattern clusters. In the conventional graph-theoretic approaches Dunn [1974] has described usage of the fuzzy graph while Chang and Pavlidis [1977] and Adamo [1980] have discussed methods to generate fuzzy decision trees.

The scope of fuzzy set theory is not limited to the classifier design. Siy and Chen [1974] used it to obtain the fuzzy descriptions, such as measures of straightness and orientation of pattern primitives.

Fuzzy-set theoretic recognition techniques can be distinguished from one another on the basis of: (1) the purpose of its usage--it can be used to describe vagueness in pattern primitives, subpatterns or pattern itself; (2) its usage in different (grammatical, graph-theoretic etc.) classification methods; and (3) methods used to compute the membership grades. Furthermore, fuzzy classification techniques differ from their non-fuzzy counterparts in some philosophical point-of-view and computational procedures.

Basic definitions used to develop the clssification procedures for both the modules are given below.

Definition 5.1

Let P be a subset of a set of patterns C, the characteristic function μ_P of P is defined as follows:

$$\mu_p(p) = \begin{cases} 1 & \text{if } p \in P \\ 0 & \text{if } p \notin P \end{cases}$$

Clearly, the characteristic function assumes the value "1" for every pattern in the subset P, so that it cannot provide much discriminatory information. If the characteristic function $\mu_p(p)$ takes a value which reflects not only the membership but also the strength of membership of each element in P then it provides more information for the classification of patterns.

To incorporate the strength of membership, consider that the characteristic function $\mu_p(p)$ may take any value in the interval [0,1]. Thus, an unknown pattern may not be identified as a pattern $p \in P$ ($\mu_p(p) = 0$), could be weakly identified as a pattern $p \in P$ ($\mu_p(p)$ near to 0), may more or less be identifiable as $p \in P$ ($\mu_p(p)$ neither too near to 0 nor too near to 1), could be strongly identifiable as $p \in P$ ($\mu_p(p)$ near to 1) or may be completely identifiable as $p \in P$ when ($\mu_p(p) = 1$). Using this imprecise definition of the characteristic function, the subset P can be considered as the fuzzy subset of the set of pattern classes C.

Definition 5.2

Given a set of pattern classes C and let $c \in C$, a fuzzy subset $\underset{\sim}{P}$ of C is a set of ordered pairs ($c|\mu_{\underset{\sim}{P}}(c)$) for all $c \in C$, where c is a class within C and $\mu_{\underset{\sim}{P}}(c)$ is a membership characteristic function that takes its value in a membership set $\Gamma = [0,1]$, which indicates the degree or level of membership of c in $\underset{\sim}{P}$.

Definition 5.3

Let $p \in \underset{\sim}{P}$ and $x \in X$, where X is some feature set, then fuzzy relation $\underset{\sim}{R}$ in P x X is defined as:

$$\text{for all } (p,x) \in \underset{\sim}{P} \times X: \mu_{\underset{\sim}{R}}(p,x) \in \Gamma = [0,1]$$

where for each pair (p,x), $\mu_{\underset{\sim}{R}}$ indicates the strength by which a feature $x \in X$ may identify an unknown pattern as $p \in \underset{\sim}{P}$.

Definition 5.4

Given a fuzzy subset $\underset{\sim}{P} = (c|\mu_{\underset{\sim}{P}}(c))$, $c \in \underset{\sim}{P}$, then entropy; $H(\zeta_{\underset{\sim}{P}}(c_1), \ldots, \zeta_{\underset{\sim}{P}}(c_n))$ is defined as (Kaufmann [1975]):

$$H(\zeta_{\underset{\sim}{P}}(c_1), \ldots, \zeta_{\underset{\sim}{P}}(c_n)) = \frac{-\sum_{i=1}^{n} \zeta_{\underset{\sim}{P}}(c_i) \,.\, \ln \zeta_{\underset{\sim}{P}}(c_i)}{\ln n}$$

where,

$$\zeta_{\underset{\sim}{P}}(c_i) = \frac{\mu_{\underset{\sim}{P}}(c_i)}{\sum_{i=1}^{n} \mu_{\underset{\sim}{P}}(c_i)}.$$

5.3 Prediction Module

Suppose that there are m classes and let $X = (x_1^r, x_2^r, \ldots, x_n^r)$ be the statistical features obtained from the training samples of the r^{th} class, where x_i^r takes its discrete values in the range 0 to L (L=4). Suppose that η_{ijk}^r is the frequency of occurrence of the j^{th} feature in the i^{th} zone (for the zone definition see section 3.2) taking a value k in the r^{th} class. Then the proportion η_{ijk}^r / N_r, where N_r is the number of training samples in the r^{th} class, can be used as the membership grade for the r^{th} class if the j^{th} feature assumes the value k in the i^{th} zone.

Using definition 5.4, the fuzzy entropy due to the j^{th} feature can be computed according to the following:

$$H_{ijk} = \frac{\sum_{r=1}^{m} \gamma_{ijk}^r . \ln \gamma_{ijk}^r}{\ln m}$$

where

$$\gamma_{ijk}^r = \frac{\xi_{ijk}^r}{\sum_{r'=1}^{m} \xi_{ijk}^{r'}}$$

and

$$\xi_{ijk}^r = \frac{\eta_{ijk}^r}{N_r}$$

H_{ijk} takes a value 1 when the j^{th} feature in the i^{th} zone with value k contributes no discriminatory information when all the classes are equally likely. The confidence factor due to the j^{th} feature assuming value k in the i^{th} zone is computed by

$$\lambda_{ijk} = 1 - \frac{K_{ijk}}{m} \times H_{ijk}$$

where K_{ijk} is the count of classes for which $\eta^{r}_{ijk} = 0$.

The confidence factor λ_{ijk} and the proportion ξ^{r}_{ijk} are used to compute the weight of a feature element taking value k in the i^{th} zone. The weight is computed as $w^{r}_{ijk} = \xi^{r}_{ijk} \times \lambda_{ijk}$. The following example illustrates the computation of the weight of a feature element.

Suppose there are 3 classes (m=3) and there are 10 samples ($N_1 = N_2 = N_3 = 10$) in each class. Furthermore, the j^{th} feature assumes the distribution ($\eta^{1}_{ijk} = 5$, $\eta^{2}_{ijk} = 6$ and $\eta^{3}_{ijk} = 1$) in the i^{th} zone for a given value of k. The proportion or the class membership ξ^{r}_{ijk} ~~by~~ which the j^{th} feature supports each class is obtained as (5/10, 6/10, 1/10). The normalized memberships γ^{r}_{ijk} are computed as (0.5/1.2, 0.6/1.2, 0.1/1.2) and the uncertainty H_{ijk} in decision making due to this feature is 0.835. Since the j^{th} feature occurs in all the three classes, K_{ijk} =3 and the confidence factor λ_{ijk} by which this feature can correctly classify an unknown is 0.165 (1-0.835). The corresponding weights w^r_{ijk} for each class are obtained as (0.0825, 0.0990, and 0.0165).

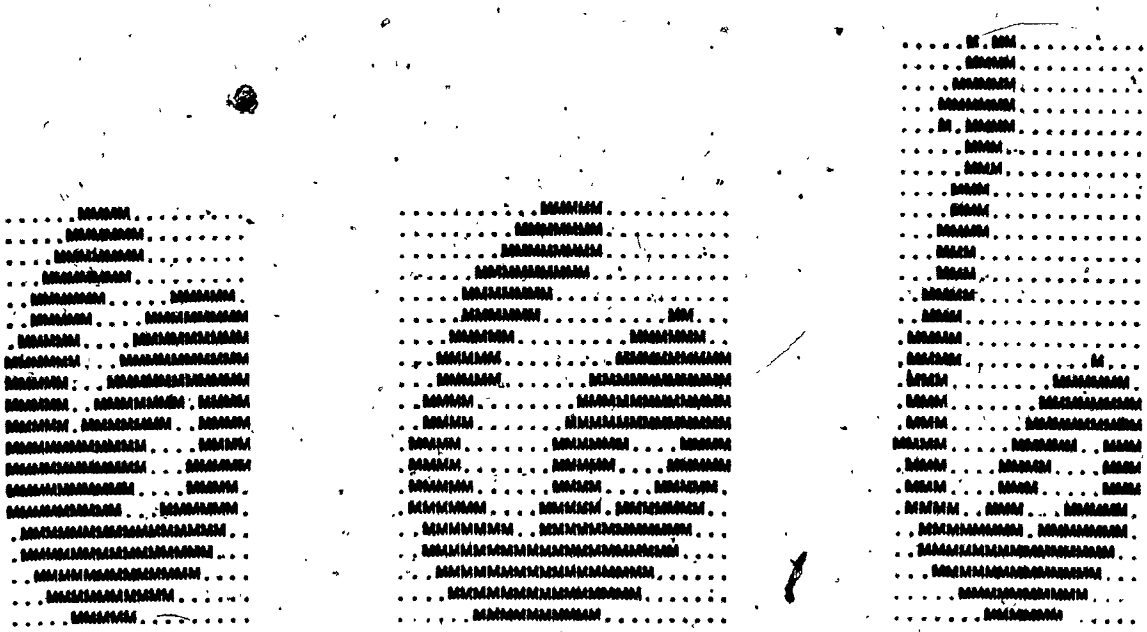
The weights w^r_{ijk} are used for prediction purposes. The prediction procedure is based on the assumption that features assuming values (0,1,2,3, or 4) independently classify an unknown pattern into a unique class. The prediction procedure is as follows:

Let u_{ijk} be the j^{th} feature extracted from the unknown pattern. $u_{ijk} = 1$ if the j^{th} feature takes the value k in the i^{th} zone; otherwise $u_{ijk} = 0$. Euclidian distances D^r_k and angles θ^r_k between the feature and weight elements for a given k in all M zones is computed according to:

$$D^r_k = \sqrt{\sum_{i=1}^{M} \sum_{j=1}^{n} (w^r_{ijk} - u_{ijk})^2}$$

$$\theta^r_k = \cos^{-1} \frac{\sum_{i=1}^{M} \sum_{j=1}^{n} w^r_{ijk} \times u_{ijk}}{\sqrt{\sum_{i=1}^{M} \sum_{j=1}^{n} (w^r_{ijk})^2} \times \sqrt{\sum_{i=1}^{M} \sum_{j=1}^{n} (u_{ijk})^2}}$$

For each value of k =0, 1, 2, ..., 4, using the distance, five predictions about an unknown were made by taking that value for which D^r_k is minimum. Similarly, other five predictions were made by considering the five minimum values of θ^r_k over all the classes r=0,1,2, ..., m. In case of a tie or whenever no feature assumes a particular value of k, the prediction for that value of k was ignored. Rules concerning the decision making process in the prediction module are discussed in Chapter six. The motivation behind introducing several predictions is the expectation that it is very unlikely that all the predictions can go wrong for a normal character image. The main purpose for the multiple prediction is to capture the implied clustering effect, that is to partition the pattern samples into expected and unexpected clusters which is illustrated in Figs. 5.2(a-c), where the expected classes are (0,6), (0,6) and (6) respectively. Classes other than these are considered as unexpected. Furthermore, these figures illustrate the power of the prediction module. For a normal image of the character "6", the prediction is unique but for the other two images prediction is not that accurate. This happens due to considerable variations in the size and shape of an image. Here different values of a feature element are used for predictions, but the same effect can be achieved by using multiple recognition schemes which are both expensive and time consuming.



(a) $\begin{Bmatrix} 0,0,0,6,6 \\ 0,0,0,6,6 \end{Bmatrix}$ (b) $\begin{Bmatrix} 0,0,6,6,6 \\ 0,0,6,6,6 \end{Bmatrix}$ (c) $\begin{Bmatrix} 6,6,6,-,- \\ 6,6,6,-,- \end{Bmatrix}$

Fig. 5.2 a-c. Implied Clustering Effect.

5.4 Structural Module

In this module a different classification strategy is adopted. This strategy was developed by analyzing the existing classification techniques. A close look at the fuzzy and/or non-fuzzy classification methods reveals that most of these methods have one thing in common, they demand a rigid definition (similar to the prediction module) of a prototype for each representative class. This problem is briefly discussed in the last Chapter. In fact, rigidity in prototype definition limits the real-time learning capability of a recognition system. Classification techniques based on cluster analysis and decision tree do provide some real-time learning capabilities, but the addition of some knowledge about an unseen pattern may require a reorganization of the system's knowledgebase. For example, consider the recognition systems based on cluster analysis, decision tree and syntax analysis. The addition of some information relevant to a new pattern may require the computation of new cluster center for the cluster analysis based systems, and reorganization of a decision tree for an optimal decision for decision tree based systems. Similarly, in syntax analysis based systems, for every unseen pattern, a new grammar needs to be constructed, which implies the modification of the syntax analyzer. These factors limit the real-time learning capabilities of a recognition system.

In our structural module a dynamic classification system is developed. Unlike conventional approaches, this approach does not require: (a) alterations (reorganization) in knowledge organization for unseen patterns; and (b) a rigid prototype definition per class. To avoid the rigidity in prototype definition, the concept of a hypothetical prototype is introduced.

We believe that a recognition system designed to meet the above two requirements will learn unseen variations in patterns in real time with greater flexibility.

5.4.1 Basic Approach

Let $C = c_1, c_2, \ldots, c_m$ be the set of m pattern classes and $X = x_1, x_2, \ldots, x_n$ be the set of n features obtained from an unknown pattern. Assume that the presence of a feature element x_i divides the pattern class C into two subsets: (1) the set of those classes in which the characteristic x_i is present; and (2) the set of those classes in which the characteristic x_i is absent.

Let P^i for all $i=1,2,3,\ldots,n$ be the subsets of the pattern class set C obtained due to the presence of feature elements x_i, for all $i=1,2,3\ldots,n$ respectively. An unknown pattern with feature set X may be a member of the non-empty subset $P = \bigcup_{i=1}^{n} P^i$ of the set C. If the cardinality of the subset P is one then an unknown pattern is uniquely identifiable. Unfortunately, this is not always the case. A feature may be present in several pattern classes. Consequently, the subset P will contain more than one element. Under this situation the set-theoretic approach cannot provide a unique identification, which is evident from definition 5.1 of the characteristic function.

Returning to the pattern classification problem, each subset $\underset{\sim}{P}^i$, for all i=1,2,3,...,n of the set C obtained due to the presence of feature elements x_i for all i=1, 2, 3...., n respectively can be considered as a fuzzy subset $(c|\mu_{\underset{\sim}{P}}(c))$, for all $c \in C$. Hence, the presence of a particular feature in an unknown pattern does not only identify an unknown as one of the members of the subset $\underset{\sim}{P}^i$, but also assigns a degree of membership. This degree of membership can be interpreted as a measure of confidence by which an unknown pattern is identified as one of the members of the set C and assigned to a subset $\underset{\sim}{P}^i$ of the set C. In special situations this characteristic membership function may take the value which is either 0 or 1, which implies that the ordinary set theory is a particular case of the theory of the fuzzy subset. Next, we generalize the theory used to classify an unknown pattern.

5.4.2 Unknown identification

The identification of an unknown is simply to compute the membership characteristic values of the fuzzy subset $P = \bigcup_{i=1}^{n} P^i$ where each P^i is a fuzzy subset $(c \mid \mu_{P^i}(c))$ for all $c \in C$, and $i=1,2,3, \ldots, n$. The membership characteristic function μ_P of the fuzzy subset P can be computed using the fuzzy relation and ideal and nonideal prototypes defined as follows:

Definition 5.5

Given a fuzzy relation R with membership characteristic function $\mu_R(p,x)$, an ideal prototype p^+ is a pattern for which

$\mu_R^+(x) = \underset{p \in P}{MAX}(\mu_R(p,x))$, for all $x \in X$

Definition 5.6

Given a fuzzy relation R with membership characteristic function $\mu_R(p,x)$, a nonideal prototype p^- is a pattern for which

$\mu_R^-(x) = \underset{p \in P}{MIN}(\mu_R(p,x))$, for all $x \in X$.

It should be noted that every feature element in X provides the maximal evidence for the ideal prototype and minimal evidence for the nonideal prototype. These prototypes can be considered as two reference points and can be created in real time from the information contributed by each feature element. It is expected that an unknown should lie within these two reference points. Using the fuzzy relation $\mu_{\underset{\sim}{R}}(p,x)$ and prototypes p^+ and p^- the membership characteristic function $\mu_{\underset{\sim}{P}}(p)$ can be computed by

$$\mu_{\underset{\sim}{P}}(p) = \frac{\psi_p^-}{\psi_p^- + \psi_p^+}, \qquad \text{for all } p \in \underset{\sim}{P}$$

where ψ_p^+ and ψ_p^- are the distances of a pattern p of $\underset{\sim}{P}$ from ideal and nonideal prototypes p^+ and p^- respectively. These distances can be computed by the relationships given below:

$$\psi_p^+ = \sqrt{\sum_{j=1}^{n} \left(\mu_{\underset{\sim}{R}}(p,x_j) - \mu_{\underset{\sim}{R}}^+(x_j)\right)^2}$$

$$\psi_p^- = \sqrt{\sum_{j=1}^{n} \left(\mu_{\underset{\sim}{R}}(p,x_j) - \mu_{\underset{\sim}{R}}^-(x_j)\right)^2}$$

The membership characteristic function $\mu_{\underset{\sim}{P}}(p)$ takes the values in the interval [0,1]. The value of $\mu_{\underset{\sim}{P}}(p)$ approaches 1 whenever ψ_p^+ approaches zero, that is, the pattern $p \in \underset{\sim}{P}$ is very similar to the ideal protoype p^+. On the other hand $\mu_{\underset{\sim}{P}}(p)$ approaches zero whenever ψ_p^- is close to 0, that is, p is very similar to the nonideal prototype p^-. The value of $\mu_{\underset{\sim}{P}}(p) = 1/2$, if $\psi_p^+ = \psi_p^-$ leads to an indeterminate situation. Thus, a simple classification procedure can classify an unknown pattern as p^*, if for all $p \in \underset{\sim}{P} \{ p^* \mid (\mu_{\underset{\sim}{P}}(p^*)) = \underset{p}{\mathrm{MAX}} \quad (p \mid \mu_{\underset{\sim}{P}}(p)) > 0.5 \}$

For the computation of the grade of membership $(\mu_{\underset{\sim}{P}}(p))$, consider the pattern shown in Fig. 5.3. The shape features (F1, F4, F7, F13, F18 and F19) obtained in both the horizontal and the vertical scans are used to form the fuzzy relation $\underset{\sim}{R}(p,x)$ see Fig. 5.3.

The characteristic membership grades for the ideal prototype $\mu^+_{\underset{\sim}{R}}(x)$ and the nonideal prototype $\mu^-_{\underset{\sim}{R}}(x)$ are obtained as (0.4600, 0.8320, 1.0, 0.9670, 0.5330, 0.7970, 0.5450, 0.5320, 0.6380) and (0., 0., 0., 0., 0., 0., 0., 0., 0.) respectively. Using these two reference points and the fuzzy distance described earlier the grade of membership $\mu_{\underset{\sim}{P}}(p)$ for each pattern is computed and it takes the following values:

$(0 \mid \mu_{\underset{\sim}{P}}(0) = 0.0214)$

$(1 \mid \mu_{\underset{\sim}{P}}(1) = 0.0006)$

$(2 \mid \mu_{\underset{\sim}{P}}(2) = 0.2646)$

$(3 \mid \mu_{\underset{\sim}{P}}(3) = 0.7836)$

$(4 \mid \mu_{\underset{\sim}{P}}(4) = 0.2097)$

$(5 \mid \mu_{\underset{\sim}{P}}(5) = 0.3092)$

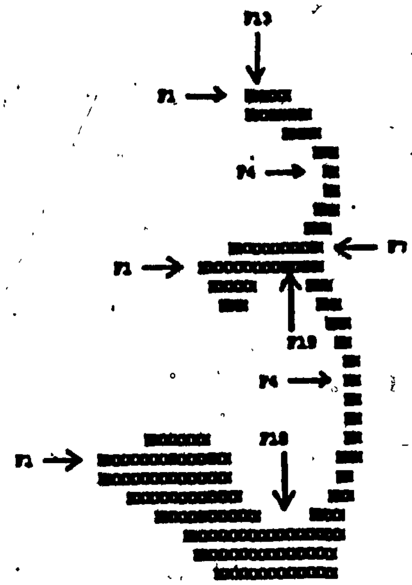
$(6|\mu_{p}(6)=0.1030)$

$(7|\mu_{p}(7)=0.0287)$

$(8|\mu_{p}(8)=0.0504)$

$(9|\mu_{p}(9)=0.0452)$

The maximum value of $\mu_{p}(p)=(0.7836)$ corresponds to the pattern class 3 and is greater than 0.5. Thus, the unknown is classified as "3".



Character Three.

Shape List.

F1 : End-point in the west direction.
F4 : Cavity with bigger lower arm facing west direction.
F7 : Cavity with bigger lower arm facing east direction.
F13: End-point in the north direction.
F18: Cavity with bigger right arm facing north direction.
F19: Cavity with bigger right arm facing south direction.

Fuzzy Relation.

Patterns	F1	F4	F1	F4	F1	F7	F13	F19	F18
	Shape Features								
0	.0027	.0000	.0063	.0000	.0256	.0000	.0392	.0000	.0000
1	.0014	.0000	.0000	.0000	.0000	.0000	.0000	.0000	.0000
2	.1900	.0000	.0190	.0000	.3870	.1330	.0494	.5320	.0000
3	.3260	.8320	1.0000	.9670	.5330	.7970	.0443	.5320	.4250
4	.0454	.0000	.0000	.0000	.0000	.0665	.5450	.0000	.0000
5	.4600	.3210	.0253	.0000	.0179	.0000	.0187	.0000	.6380
6	.0000	.0000	.0063	.0000	.0000	.0000	.2440	.0000	.0000
7	.0254	.0000	.0000	.0000	.0564	.0000	.0153	.0000	.0000
8	.0053	.0000	.0063	.0000	.0256	.0665	.0869	.0000	.0000
9	.0080	.0000	.0000	.0967	.0179	.0000	.0204	.0000	.0000

Fig.6.3. Shape Features and Fuzzy Relation.

5.4.3 Implementation

For every shape primitive extracted from an unknown pattern, its location is used as the key to search its best fit location in the coressponding shape relation (sec. 3.5). Searching is done by computing the city block distance between the location of the best fit rectangle of the shape from an unknown pattern and the location of the same shape primitive in the database. The tuple for which the computed distance is minimum is selected. The frequency, η_r , r=1, 2, 3, ..., m of occurrence of the shape primitive stored in that tuple during the training session (see sec 3.5) is used to construct the fuzzy relation $\underset{\sim}{R}$. The membership characteristic function $\mu_{\underset{\sim}{R}}$ (p,x) where x denotes the underlying shape primitive and p corresponds to the pattern classes in which that shape has been observed, is computed as η_r / N_r for all r =1, 2, 3, ..., m.

Using the method described in the last section the membership function $\mu_{\underset{\sim}{P}}$ (p) is computed and the unknown pattern is identified. The performance of this scheme is presented in the next Chapter. A flowchart depicting the complete recognition process can be found in Appendix C.

Chapter 6

Test Data and Experimental Results

6.1 Test Data

The test data set used in this research was collected at fourteen dead letter mail offices throughout the United States by the U. S. Postal Service, Research and Development Department, Pattern Recognition and Communication Branch. The basic requirements for accepting mail pieces in the database were that the address was handwritten, that the ZIP code was not defaced and did not occupy an area in excess of 9.75 mm x 34.14 mm. Each ZIP code was digitized on a grid of 64 x 224, 0.153 mm square elements. This corresponds to a resolution of 166 dots per inch.

ZIP codes were microfilmed with a slightly bluish green spectral response (530 nm) to prepare the positive transparencies of each ZIP code image. A 1.615 micron, programmable, CRT scanner (Information International Inc. PFR-3) scanned the ZIP codes from the positive transparencies. After intermediate processing (64 density levels), the scanner output was finally recorded at 16 reflectance levels. The filming reduction ratio (2.028:1) and the subsequent data reduction resulted in 0.0054 inch as the effective scanning spot size. The effective spot was derived from the 25 equally spaced scanner spot in 5 x 5 arrays, resulting in the 5.4 mil approximately square spot. The encoding of the grey level ranges from 0 to 15 where 0 represents the darkest point and "F" represents the lightest point (Fig. 6.1).

The data pose a variety of problems faced in real life applications like: (1) the paper quality (rough, coarse matte, linen, smooth, glossy, ribbed, ridges and others), paper colors ranging from white to black, paper patterns (none, straight, figured, shaded and others); (2) ink colors (blue, black, green, red, brown and others); (3) pen types (quill, ballpoint, felt tip, pencil and others) and (4) writing style and random distortions such as field intrusions, skipping, right and left tappers, variations, skewness, open loops, nesting, write-overs, overlap and touching, etc., (Fig. 6.2).

It should be noted that we have not tried to solve the problem of skipping in this study. All other cases were handled during the binarization and segmentation stages described in Chapter two.

The ZIP codes were automatically segmented under human supervision. From each ZIP code image the individual digits were extracted and displayed along its labels for the label verification. Incorrect labels were manually corrected and separate files of each digit were prepared.

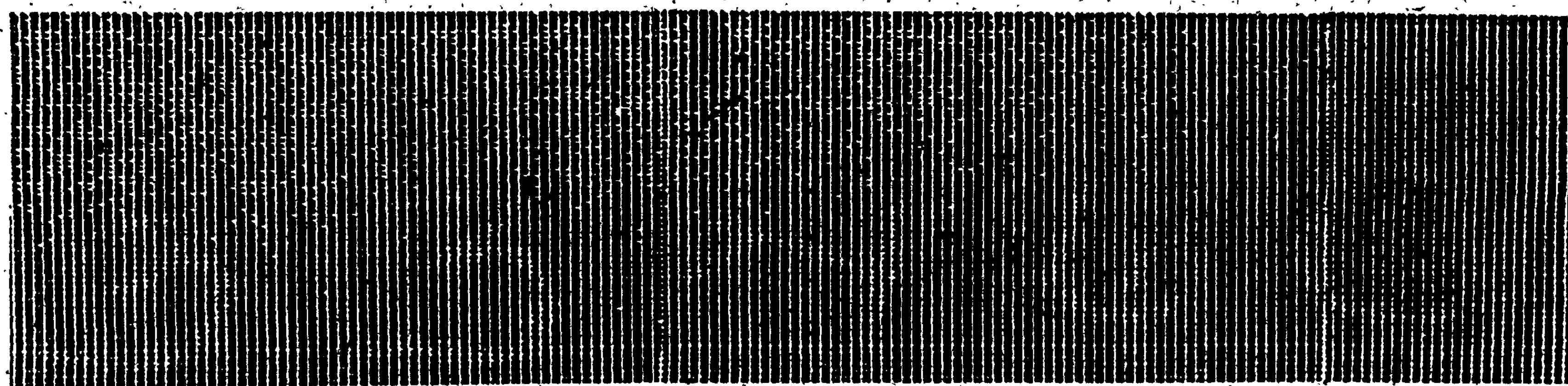


Fig. 6.1. A 16 Grey Level ZIP Code Image 33460.

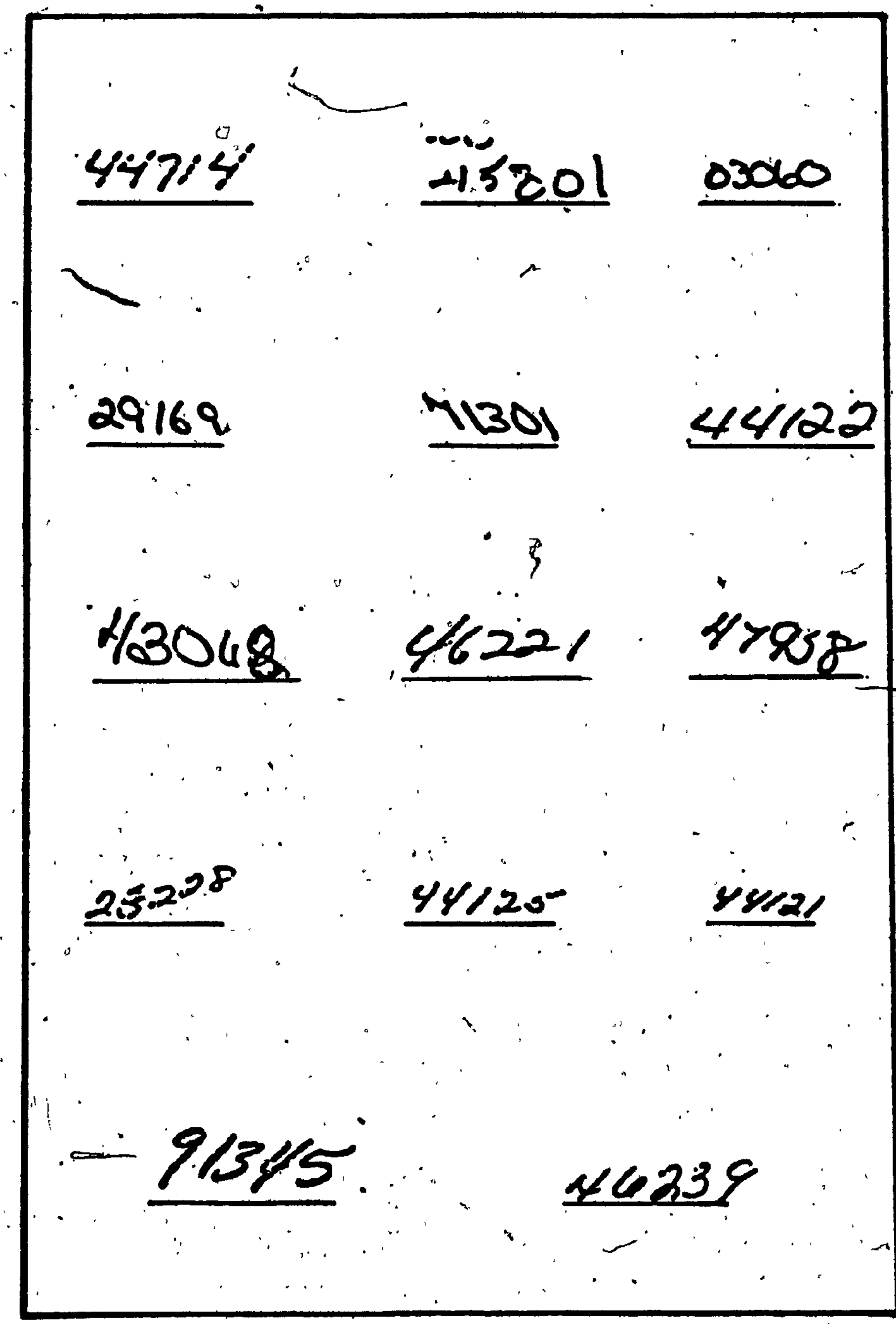


Fig. 6.2. Some Samples of Original ZIP Codes.

6.2 Experimental Results

The results presented in this section are mainly concerned with the recognition of individual digits rather than the ZIP codes. No attempt is made to study the recognition performance on the entire ZIP codes due to the problems faced in the segmentation process. Though the segmentation procedure performed well in ideal conditions it failed sometimes in some cases such as skipping, touching and missing digits. Due to these cases all the five symbols of a ZIP code were not always available in good shape for the recognition module. On the basis of our experience on such field data, we expect that approximately 50% ZIP codes can be correctly segmented. The major reasons for this low percentage are--the poor quality of the images and the problems highlighted in section 6.1.

The experimental data set consists of 8540 digit images selected from the ZIP code database. During the training session of the prediction module the classifier was trained in steps of 200 samples (20 samples/class). One of the reasons to train the classifier in steps is to study its prediction behaviour for both the seen and unseen patterns. After each training step two tests were performed, one to test the prediction ability of this module on the data already included in the training set and the other to test the same on the data not included in the training set. For both tests, a set of 200 samples were randomly selected from the training set and outside the training set. The prediction capability (at least one correct prediction out of 10) is illustrated in Figs. 6.2.1(a-b). The main purpose of this study was to determine the training set size for this module. Figs.6.2.1(a)-(b) show that after training on the 2000 samples, the prediction for both the training and testing data fluctuates in the range of 94% to 98%. The training was stopped after 5000 samples (500 samples per class) because no significant change in the prediction behaviour was observed.

It should be noted that in the beginning the correct prediction percentage for the training set was high but it gradually declined. Similarly, for the testing set, initially the correct prediction was low but it gradually improved as training progressed. In the case of training set this happens due to the randomness of the data set (e.g. size and shape variations). In the test case initially the performance improved because some discriminatory features may not occur in the initial training set. As training progressed, some additional features related to the different shapes of characters of the same class (e.g., 2, 2, 2, ...) are added to the knowledgebase of the system.

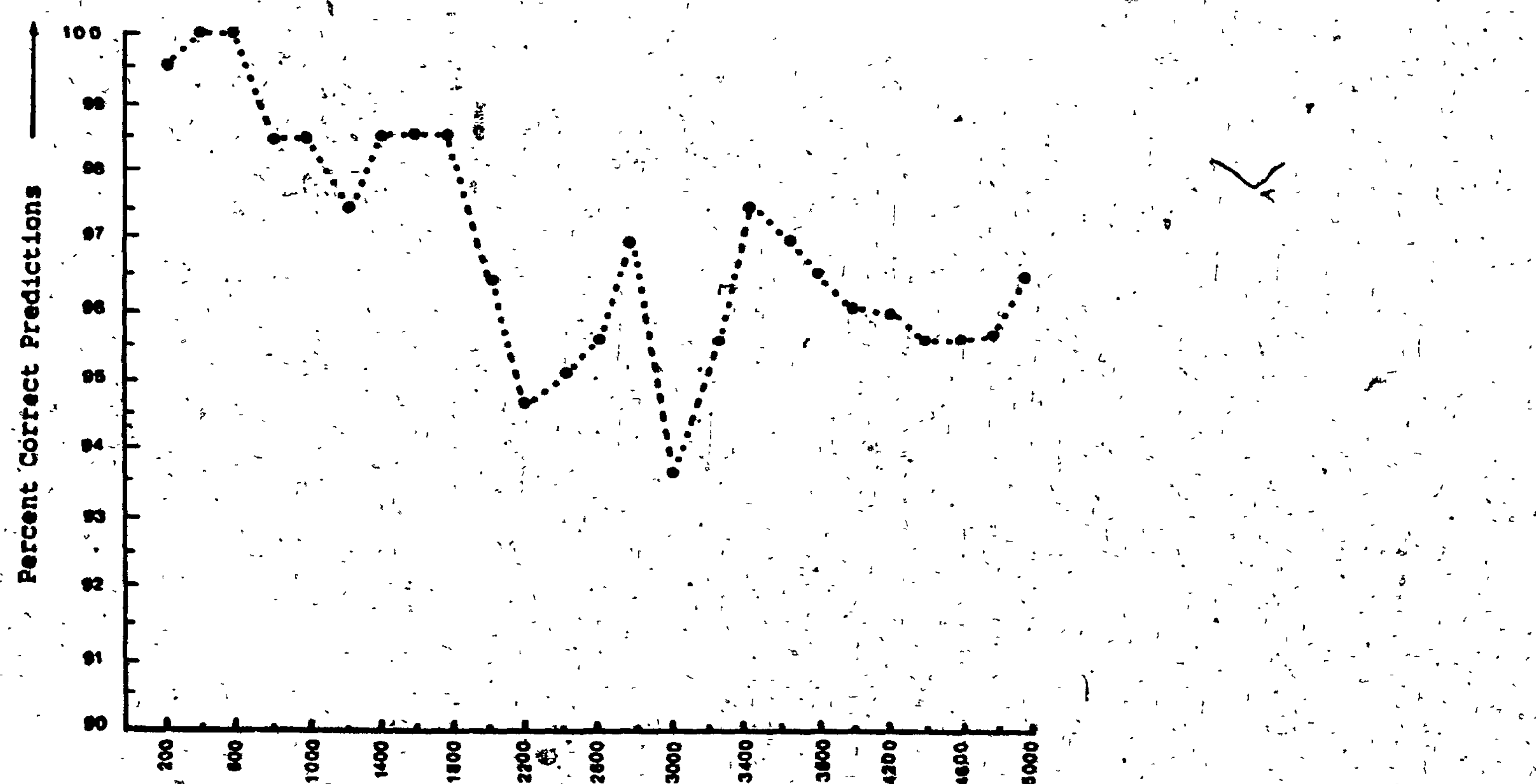


Fig. 6.2.1 a. Prediction Performance on the Training Set.

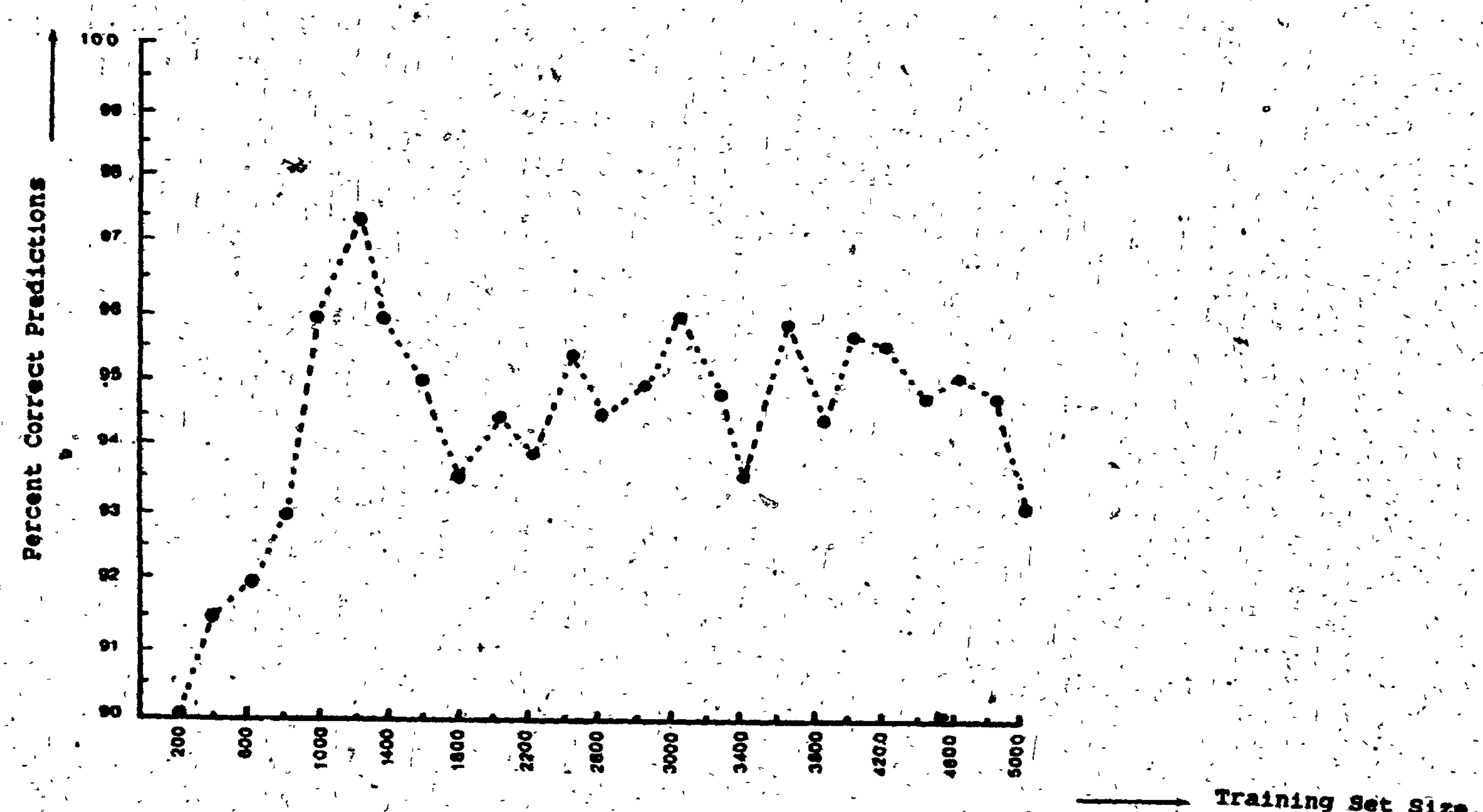


Fig. 6.2.1 b. Prediction Performance on the testing set.

6.2.1 Recognition Module

Three basic decision making processes, namely, statistical, structural and hybrid (statistical and structural combined) were studied. For the statistical method the information gathered in the prediction module is used. The following rules were used to study the performance of the decision making process based on the numerical features.

Rule 1.

Unique decision, that is, all the active decisions from the 10 predictions are unanimous i.e. the underlying unknown pattern is classified into a unique class.

Other rules are based on the cases where an unknown pattern cannot be classified by Rule 1, that is, the active decisions split and the unknown pattern is classified into more than one class.

Rule 2.

All the active decisions place an unknown pattern into two different classes.

Rule 3.

All the active decisions place an unknown pattern into three different classes.

Rule 4.

All the active decisions place an unknown pattern into four different classes.

Rule 5.

All the active decisions place an unknown pattern into five different classes.

In rules 2-5 a simple majority rule was applied to recognize an unknown pattern. In the case of tie or when all the active decisions place an unknown pattern into more than five classes, the unknown pattern is rejected. The performance statistics of the system on the testing and training sets are shown in Figs. 6.2.2 (a-b) respectively. The confusion table fot each rule on the testing and training set is given in Appendix A. Cumulative (using all the rules (1-5)) confusions for both the training set (5000 samples) and testing set (3540 samples) are shown in Tables 6.2.1 (a-b).

Table No. 6.2.1 (a). Training Set.
Prediction Module
Using Rule 1 To Rule 5

	0	1	2	3	4	5	6	7	8	9	Samples Processed	Samples Rejected	Samples Substituted	Samples Per class
0	469	3	0	0	0	2	1	1	6	0	482	18	13	500
1	0	472	5	4	0	0	0	1	8	0	490	10	18	500
2	11	5	342	32	0	7	5	15	19	1	437	63	95	500
3	2	4	2	460	0	4	0	2	10	3	487	13	27	500
4	5	12	0	0	394	1	2	1	4	53	472	28	78	500
5	8	3	0	70	7	351	4	0	7	13	463	37	112	500
6	9	8	0	0	0	12	448	0	6	0	483	17	35	500
7	5	9	0	0	5	1	0	445	5	7	477	23	32	500
8	0	7	2	35	1	4	0	2	407	18	476	24	69	500
9	2	5	1	5	8	0	1	7	12	426	467	33	41	500
	511	528	352	606	415	382	461	474	484	521	4734	266	520	5000

Recognition Reliability = 89.02%

Recognition Rate = 84.28%
Substitution Rate = 10.40%
Rejection Rate = 5.32%

Table No. 6.2.1 (b). Testing Set.
Prediction Module
Using Rule 1 To Rule 5

	0	1	2	3	4	5	6	7	8	9	Samples Processed	Samples Rejected	Samples Substituted	Samples Per class
0	617	2	1	2	1	3	4	0	12	0	642	18	25	660
1	1	611	11	4	0	0	0	4	10	1	642	18	31	660
2	5	1	99	16	1	2	2	8	7	0	141	19	42	160
3	1	0	0	137	0	3	1	2	10	1	155	5	18	160
4	12	5	0	1	234	1	3	1	5	43	305	15	71	320
5	0	0	0	6	0	79	2	0	5	4	96	4	17	100
6	27	7	1	12	4	12	480	0	12	1	556	44	76	600
7	17	15	0	3	8	1	0	504	8	10	566	34	62	600
8	1	2	1	12	1	1	0	1	94	8	121	19	27	140
9	0	4	1	0	4	0	0	3	3	116	131	9	15	140
	681	647	114	193	253	102	492	523	166	184	3355	185	384	3540

Recognition Reliability = 88.55%

Recognition Rate = 83.92%
Substitution Rate = 10.85%
Rejection Rate = 5.23%

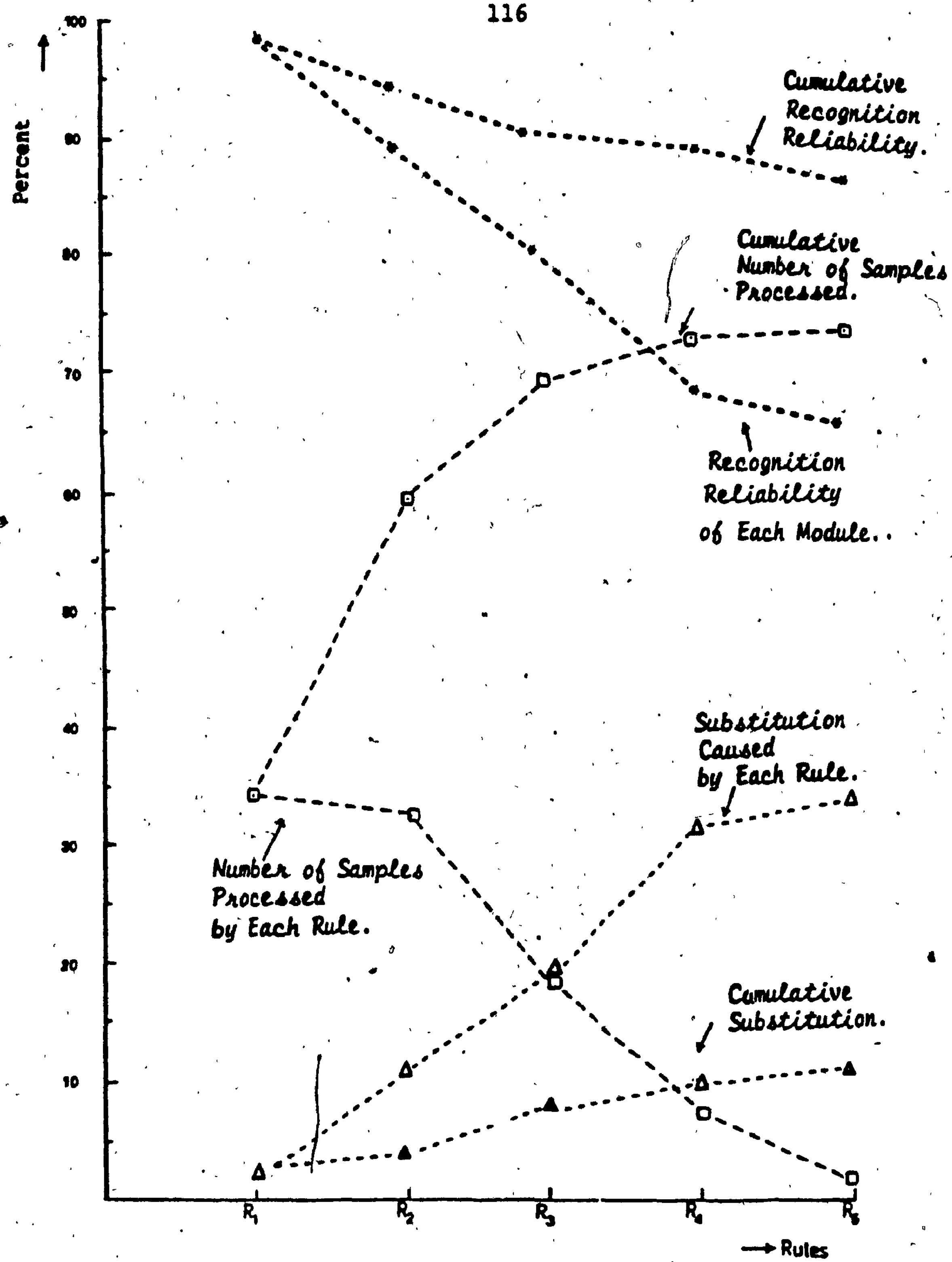


Fig. 6.2.2 a. Recognition Performance--Training Set.

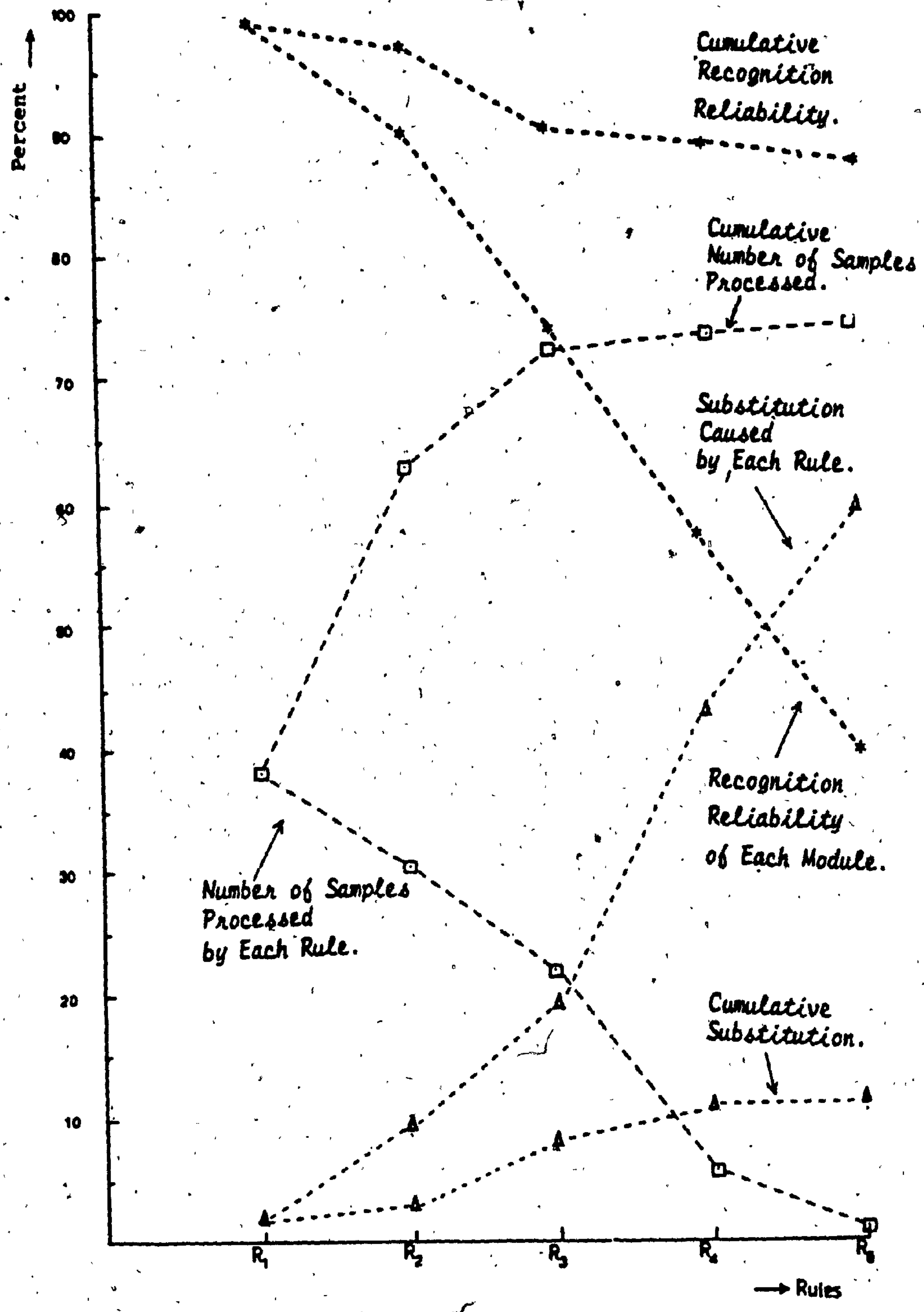


Fig. 6.2.2 b. Recognition Performance--Testing Set.

For the structural module, the classifier was trained on the same 5000 samples (500 samples/class) as used in the statistical module. The perfomance of this approach was tested on the data rejected by both rule 1 and rule 2 and is displayed in the confusion Tables 6.2.2 (a-b). Comparing the recognition reliability and rejection rate of this approach with the statistical approach, we find that this approach performs much better than the statistical approach based on rules 3-5 (Appendix A).

In the hybrid approach the recognition rates attained using rules 1 and 2 in the statistical approach Tables 6.2.3(a-b) and structural approach Tables 6.2.2 (a-b) are used to demonstrate the capability of the system. The performance of this approach is shown in the Tables 6.2.4 (a-b) which demonstrate significant improvements over the statistical approach Tables 6.2.1 (a-b).

It is evident from the results that the combination of the structural analysis with rigid acceptance rule in the statistical analysis phase will yield a better recognition score. But in this case approximately 60 to 70 percent of the data will be proccessed by the structural analysis module which is an expensive process.

Table No. 6.2.2 (a) Training Set.
Structural Analysis

	0	1	2	3	4	5	6	7	8	9	Samples Processed	Samples Rejected	Samples Substituted	Samples Per class
0	165	0	0	0	0	0	0	0	0	0	165	12	0	177
1	0	49	0	0	0	0	0	0	0	0	49	0	0	49
2	0	0	248	0	0	0	0	0	0	0	248	33	0	281
3	0	0	0	132	0	0	0	0	0	0	132	7	0	139
4	0	0	0	0	158	0	0	0	0	0	158	5	0	163
5	0	0	0	0	1	176	0	0	0	0	177	7	1	184
6	0	0	0	0	0	1	168	0	0	0	169	10	1	179
7	0	0	0	0	0	0	0	143	0	0	143	13	0	156
8	0	0	0	0	0	0	0	0	157	0	157	13	0	170
9	0	0	0	0	1	0	0	0	0	139	140	18	1	158
	165	49	248	132	160	177	168	143	157	139	1538	118	3	1656

Recognition Reliability = 99.80%

Recognition Rate = 92.69%
Substitution Rate = .18%
Rejection Rate = 7.13%

Table No. 6.2.2 (b) Testing Set.
Structural Analysis

	0	1	2	3	4	5	6	7	8	9	Samples Processed	Samples Rejected	Samples Substituted	Samples Per class
0	146	0	0	0	1	0	1	0	1	0	149	27	3	176
1	0	62	0	0	0	0	0	0	0	0	62	11	0	73
2	0	0	78	0	0	0	0	0	0	0	78	16	0	94
3	0	0	0	34	0	0	0	0	0	0	34	12	0	46
4	0	0	0	0	91	0	0	0	0	0	91	13	0	104
5	0	0	0	1	0	36	0	0	0	0	37	2	1	39
6	0	1	0	0	0	1	169	0	1	0	172	78	3	250
7	0	0	0	0	0	0	0	151	0	0	151	61	0	212
8	0	0	0	0	0	0	0	0	54	0	54	15	0	69
9	0	0	0	0	1	0	0	0	0	26	27	13	1	40
	146	63	78	35	93	37	170	151	56	26	855	248	8	1103

Recognition Reliability = 99.06%

Recognition Rate = 77.5%
Substitution Rate = .73%
Rejection Rate = 22.48%

Table No. 6.2.3 (a). Training Set.
Prediction Module
Using Rule 1 To Rule 2

	0	1	2	3	4	5	6	7	8	9	Samples Processed	Samples Rejected	Samples Substituted	Samples Per class
0	322	0	0	0	0	0	0	0	1	0	323	177	1	500
1	0	446	2	0	0	0	0	0	3	0	451	49	5	500
2	1	1	192	13	0	2	0	6	4	0	219	281	27	500
3	0	0	0	358	0	0	0	0	3	0	361	139	3	500
4	2	3	0	-0	299	0	0	0	0	33	337	163	38	500
5	1	0	0	40	3	263	1	0	2	6	316	184	53	500
6	2	2	0	0	0	7	309	0	1	0	321	179	12	500
7	2	6	0	0	1	0	0	331	2	2	344	156	13	500
8	0	3	0	18	0	1	0	0	302	6	330	170	28	500
9	1	3	0	2	4	0	0	4	1	327	342	158	15	500
	331	464	194	431	307	273	310	341	319	374	3344	1656	195	5000

Recognition Reliability = 94.17%
Substitution Rate = 3.90%
Rejection Rate = 33.12%

Table No. 6.2.3 (b). Testing Set.
Prediction Module
Using Rule 1 To Rule 2

	0	1	2	3	4	5	6	7	8	9	Samples Processed	Samples Rejected	Samples Substituted	Samples Per class
0	478	1	0	1	0	0	0	0	4	0	484	176	6	660
1	0	577	4	0	0	0	0	1	5	0	587	73	10	660
2	0	0	57	6	0	0	0	2	1	0	66	94	9	160
3	1	0	0	106	0	1	0	1	4	1	114	46	8	160
4	2	2	0	0	186	0	1	0	0	25	216	104	30	320
5	0	0	0	4	0	51	2	0	3	1	61	39	10	100
6	2	2	0	2	1	2	340	0	1	0	350	250	10	600
7	5	7	0	0	3	0	0	370	1	2	388	212	18	600
8	0	1	0	7	1	0	0	0	61	1	71	69	10	140
9	0	1	0	0	1	0	0	0	1	97	100	40	3	140
	488	591	61	126	192	54	343	374	81	127	2437	1103	114	3540

Recognition Reliability = 95.32%
Substitution Rate = 3.22%
Rejection Rate = 31.16%

Table No. 6.2.4(a). Training Set.
Structural Analysis And Rule 1 And Rule2

	0	1	2	3	4	5	6	7	8	9	Samples Processed	Samples Rejected	Samples Substituted	Samples Per class
0	487	0	0	0	0	0	0	0	1	0	488	12	1	500
1	0	495	2	0	0	0	0	0	3	0	500	0	5	500
2	1	1	440	13	0	2	0	6	4	0	467	33	27	500
3	0	0	0	490	0	0	0	0	3	0	493	7	3	500
4	2	3	0	0	457	0	0	0	0	33	495	5	38	500
5	1	0	0	40	4	439	1	0	2	6	493	7	54	500
6	2	2	0	0	0	8	477	0	1	0	490	10	13	500
7	2	6	0	0	1	0	0	474	2	2	487	13	13	500
8	0	3	0	18	0	1	0	0	459	6	487	13	28	500
9	1	3	0	2	5	0	0	4	1	466	482	18	16	500
	496	513	442	563	467	450	478	484	476	513	4882	118	198	5000

Recognition Reliability = 95.94%

Recognition Rate = 93.68%
Substitution Rate = 3.96%
Rejection Rate = 2.36%

Table No. 6.2.4(b). Testing Set.
Structural Analysis And Rule 1 And Rule 2

	0	1	2	3	4	5	6	7	8	9	Samples Processed	Samples Rejected	Samples Substituted	Samples Per class
0	624	1	0	1	1	0	1	0	5	0	633	27	9	660
1	0	639	4	0	0	0	0	1	5	0	649	11	10	660
2	0	0	135	6	0	0	0	2	1	0	144	16	9	160
3	1	0	0	140	0	1	0	1	4	1	148	12	8	160
4	2	2	0	0	277	0	1	0	0	25	307	13	30	320
5	0	0	0	5	0	87	2	0	3	1	98	2	11	100
6	2	3	0	2	1	3	509	0	2	0	522	78	13	600
7	5	7	0	0	3	0	0	521	1	2	539	61	18	600
8	0	1	0	7	1	0	0	0	115	1	125	15	10	140
9	0	1	0	0	2	0	0	0	1	123	127	13	4	140
	634	654	139	161	285	91	513	525	137	153	3292	248	122	3540

Recognition Reliability = 96.29%

Recognition Rate = 89.54%
Substitution Rate = 3.45%
Rejection Rate = 7.01%

6.3.1 Comparison

It is virtually impossible to compare the recognition performances of different recognition systems, because of the diversified range of differences in experimental procedures and the quality of the test data which plays a central role in a recognition system. However, for illustration purposes some results on handprinted numerals from the field data are given below.

Using a combination of statistical and syntactical approaches, Duerr et al [1980] conducted an experiment on handwritten numerals. They used a lab prepared data set consisting of 10,000 numerals written by approximately 200 persons. Their system was trained on 5000 numerals, and regarding the overall performance they state: "if no samples are rejected a recognition rate of 99.5% is obtained".

In the experiment conducted by Spanjersberg [1976] the data were obtained from more than 200,000 randomly chosen account holders who wrote the numbers in their own styles in specified areas (constrained data). In the training stage, a recognition rate of 94% was obtained. In the testing stage, only 60% of the payment orders (documents) were correctly identified with 0.01% and 39.99% of substitution and rejection rates respectively.

In another experiment conducted by Neill [1969] on the ZIP code data collected from mail samples written by individuals of different nationality, a recognition rate of 85% on the 5000 training samples was reported.

It is practically impossible to program each method and compare the results with our scheme. However, to demonstrate the effectiveness of this scheme, experiments were conducted using the most effective characteristic loci features. Several types of numerical and structural features have been used in handprint recognition (Suen [1980]). The multi-directional Loci (Suen, [1982a]) were chosen as a standard for the reasons: (1) It has been studied by several research groups; (2) It is relatively insensitive to noise, breaks and stroke variations; and (3) Parallel process of detecting features is relatively simple. In these experimentations, a clustering based recognition scheme was used and the results on the selected samples from the ZIP code database are summarized in Table 6.3 on the next page (Tien [1985]).

These results were obtained with very few samples and it is anticipated that further improvements may be made once the system has been trained with more samples. Our results on the testing set using the hybrid approach (Table 6.2.4b) shows an improved performance of between 4-18%; our testing was also done on a substantially larger and randomly chosen data set.

Table 6.3.

	Correct	Error	Reject
Training set 390 samples	84.62%* 99.23%**	0.0%* 0.7%**	15.38%* 0.00%**
testing set 333 samples	71.77%* 85.59%**	4.2%* 10.5%**	24.20%* 3.90%**

* Recognition Results using Multi-directional Loci.
** Recognition Results Using Multi-directional Loci and additional features related to the center of gravity, zones etc.

The performance of our method may be compared with some recent works--particularly with the syntactical approaches. In the study on the 400 handprinted numerals (lab prepared data), Shridhar and Badreldin [1985] reported a recognition accuracy of 99%. A direct comparison of their scheme with ours cannot be made. However, both schemes may be compared on the basis of the recognition models and their use in real-life applications. As a recognition model, Shridhar et al have used an automaton whose state transitions were derived from a small set of samples--which may not necessarily cover unexpected variations in handwritings and does not provide a formal mechanism to study handwriting variations. Thus, as mentioned in Chapter 4, like other syntactical approaches to character recognition, their model is rigid and requires modification of the automaton for variations perceived in the future stages. In our approach such variations can be easily incorporated in the knowledgebase by adding separate tuples for the structural anlysis or re-calculation of the confidence factors in the prediction module. For practical applications, in terms of recognition reliability, our model has shown better performance on a much larger set of field data, especially the structural analysis where substitution error is well under 1% and reliability is over 99%.

Chapter 7

Performance Analysis, Future Work and Conclusion

7.1 Performance Analysis

In this section first the results of our experiment are discussed and finally typical rejected and substituted samples are examined.

7.1.1 Discussion

As mentioned earlier, character recognition is a challenging field of research, and it is the basis of digital document processing. Requirements in the office automation have created, more than ever, needs for the development of reliable digital document processing equipment. Though, during the past few decades, different approaches have been applied to solve the character recognition problem, even good results were reported, their reliability and practical applicability cannot be judged unless these recognition models are tested with field data. In this research the applicability of popular recognition approaches, namely, the statistical and the structural recognition schemes on the field data is explored. It should be noted that majority of research works in character recognition (in general pattern recognition) either a statistical or a structural approach is implemented and tested. In our research, separate statistical and structural techniques are developed and tested. Also a method to combine these techniques is examined. Performance of these techniques is summarized in Table 7.1, which shows individual results of: (a) the statistical approach--obtained by applying rules 1 to 5; (b) the structural approach--obtained by performing the structural analysis on the samples rejected by rules 1 and 2 and (c) the hybrid approach--obtained by considering the results of rules 1 and 2 and the structural approach. The performances of each rule and the combination of rules 1 and 2 are shown in Table 7.2.

The substitution rates for rule 1 in the prediction module (Table 7.2) and the structural module (Table 7.1) are less than one percent. Using rule 1 and the structural approach the same performance as with the structural approach alone could be obtained. However, it should be noted that under this combination only 30% of the data will be processed by the statistical approach and the remaining data will be processed by the structural approach but at the expense of higher computational cost.

Recognition Schemes	Training set				Testing set			
	Relia-bility	Substi-tution	Rejec-tion	Sample Size	Relia-bility	Substi-tution	Rejec-tion	Sample Size
Statistical	89.02%	10.40%	5.32%	5000	88.55%	10.85%	5.23%	3540
Structural	99.80%	0.18%	7.13%	1656	99.06%	0.73%	22.48%	1103
Hybrid	95.94%	3.96%	2.36%	5000	86.2[illegible]%	3.45%	7.01%	3540

Table 7.1. Performance Summary

Rules Used	Training set				Testing set			
	Relia-bility	Substi-tution	Rejec-tion	Number of Samples	Relia-bility	Substi-tution	Rejec-tion	Number of Samples
Rule 1	98.38%	.56%	65.48%	5000	98.81%	.45%	61.89%	3540
Rule 2	89.68%	5.10%	50.58%	3274	90.99%	4.47%	50.34%	2191
Rule 3	80.48%	11.23%	42.45%	1656	77.09%	13.69%	40.25%	1103
Rule 4	68.56%	16.50%	47.51%	703	56.31%	21.85%	50.00%	444
Rule 5	66.18%	6.89%	79.64%	334	40.54%	9.91%	83.33%	222
Rule 1 and Rule 2.	94.17%	3.90	33.12%	5000	95.32%	3.22%	31.16%	3540

Table 7.2. Recognition Performance of different Rules.

7.1.2 Rejection and Substitution

Typical rejected and substituted samples are shown in Figs. 7.1 and 7.2 respectively. Rejected samples are examined first. An unknown is rejected--(i) if its membership characteristic, computed in the structural module is less than 0.5 or (ii) if the identification by the structural module is not the same as the frequent member (majority prediction) of the predicted class.

Referring to Fig. 7.1, characters "0", "2", "3", "5", "6", and "8" were rejected under the first criterion and "1", "4", "7" and "9" were rejected under the second criterion.

Our observation shows that there are three major reasons for substitutions: (i) random errors--broken samples were frequently substituted for some unexpected class, for example, broken five's (Fig. 7.2 (d)) were substituted for a three; (ii) size variations--samples having less than half of the normalized size were assigned to a random class, for example, a zero shown in Fig. 7.2 (a) is recognized as four and (iii) feature dominance--infrequent samples of a class having features in common with some frequent classes. A perfect example is the character 6 shown Fig. 7.2 (f), where the cavity facing the right is in the position of the similar cavities frequently appearing in five's, and the bottom hole feature which is very frequent in samples of six and eight and less frequent in five's, three's and two's were dominated by that cavity feature. Consequently this six was substituted by five.

The random errors are difficult to control. However, substitution errors may be minimized by considering suitable normalization schemes to counter size variations. And, errors due to feature dominance can be minimized by incorporating some additional knowledge, such as, rules which can eliminate ambiguities. For example, incorporating a rule stating that a pattern without left cavity cannot be accepted as five will not permit such errors. A general set of such rules can only be derived by a manual study of large failure set and is a topic for future research (see Section 7.2 below).

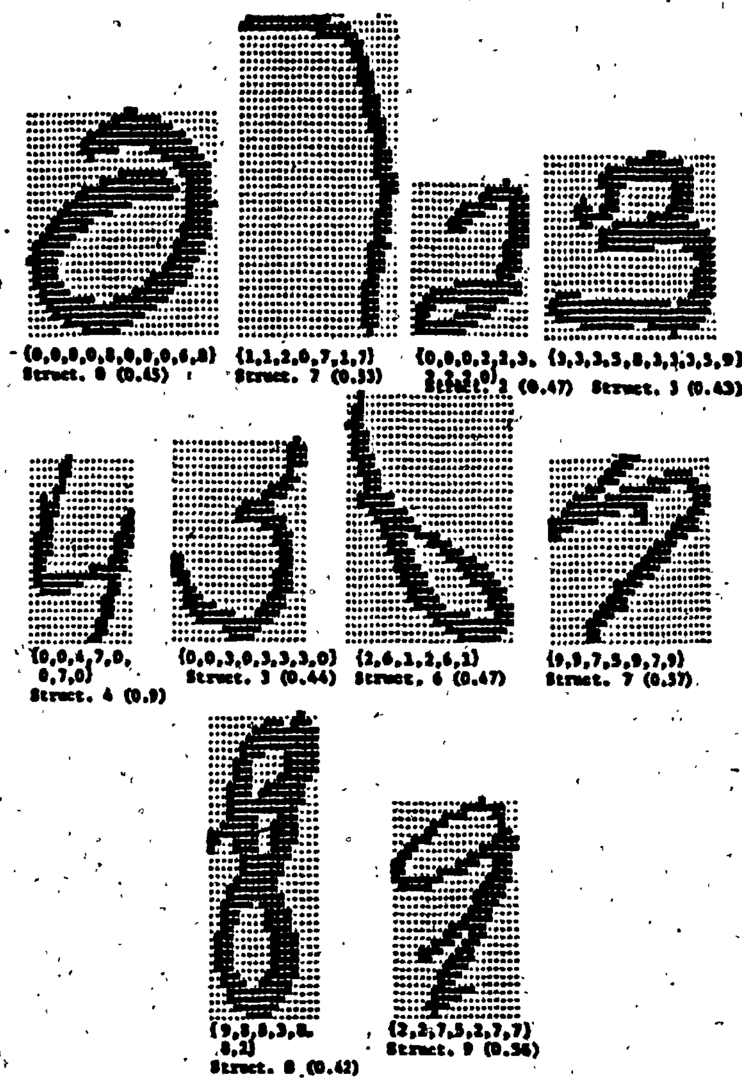


Fig. 7.1. Typical Rejected Samples 0-9

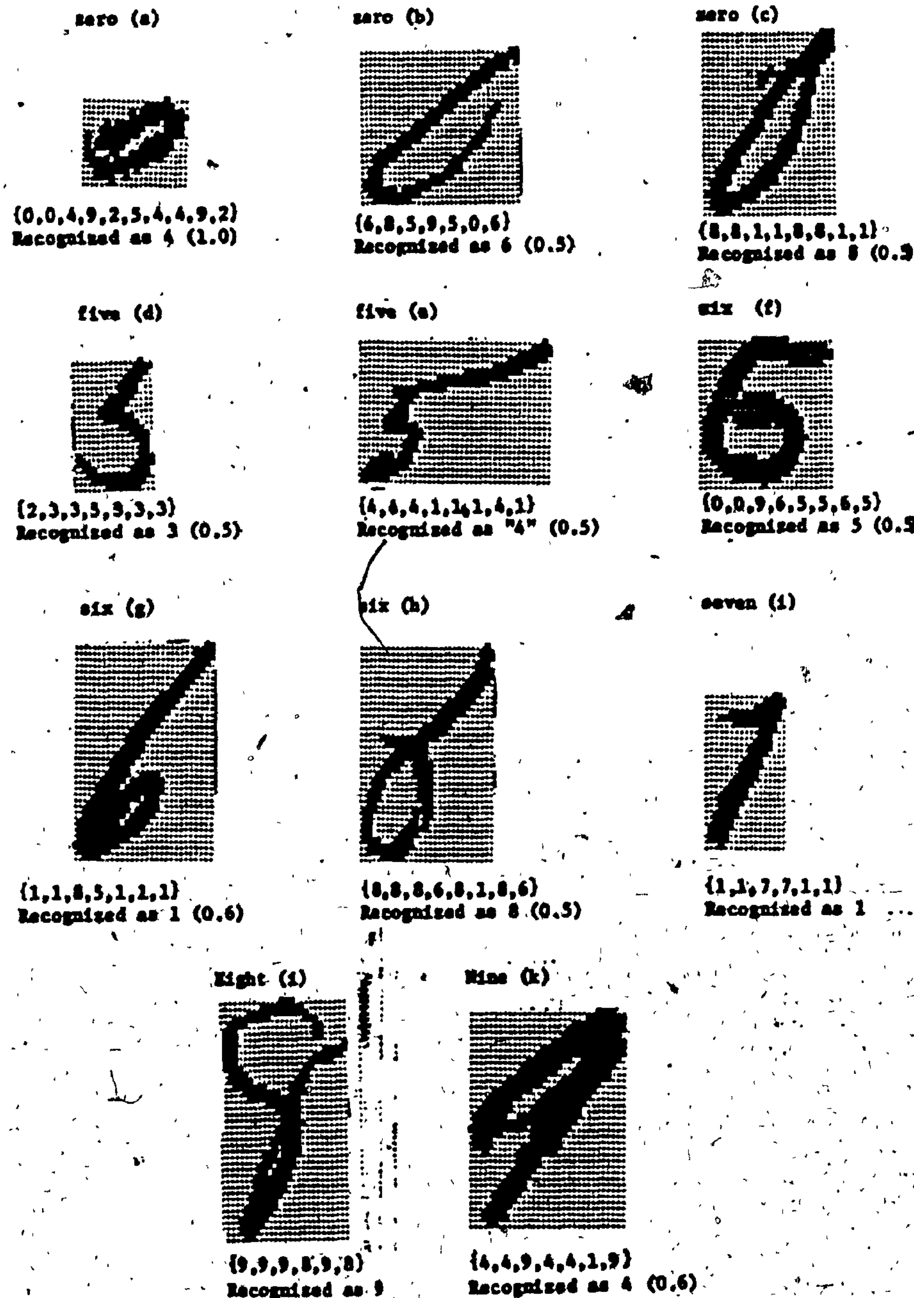


Fig. 7.2. Typical Substituted Samples.

7.2 Future Research

Research reported here can be extended in three major directions: (a) higher order shape extraction--as illustrated in Chapter three, the shape extraction method can be extended to define and extract more complex shape properties (second, third order cavities) useful for the analysis of complex patterns, for example, characters from other languages (Krishnamoorthy et al [1983]; Ahmed and Suen [1984]; Suen [1986]) and general patterns like map analysis (Ahmed et al [1985]); (b) special purpose hardware--parallelism involved in the prediction and the structural modules can be realized by means of the special purpose hardware. Applicability of such architectures in pattern recognition (in particular character recognition) is reviewed in (Siddiqui and Ahmed [1984]). (c) development of a rule-based classification scheme--ambiguities such as those described in the last section and other types can be removed by discovering rules during the training phase of a recognition system.

7.3 Conclusion

Generally speaking, research works in character recognition are considered new if they present some new ideas on feature definition, feature extraction, feature selection or character classification. The image processing and segmentation part which is essential for a reading machine and can affect the feature extraction and classification is frequently ignored or rarely mentioned. In this research, a complete model of a machine capable of reading totally unconstrained handwritten numerals, covering required image processing-- enhancement and binarization, a new segmentation approach, both the statistical and new structural feature definition and extraction and a two stage classification method has been developed and tested on the huge field data.

The image processing part developed for this research is problem dependent but our segmentation and structural feature extraction methods is very general and its applications are not limited to text or character patterns. It can be applied to any binary pattern, particularly, the structural feature extraction scheme which does not depend on a priori shape definition, instead it allows to define shapes in terms of a finite set of fundamental shapes. In addition, the relational database model provides a convenient mechanism for the storage and retrieval of the information related to a shape and its attributes.

The motivation behind the development of several classification schemes was the search for a better, practical and minimum error classifier. The prediction module was developed with intentions to judge the quality of an unknown pattern, and induce the feeling of doubt as we human experience while reading poor samples of handwritings. The structural module was developed as an aid to deal with dubious situations.

Through extensive experiments, it has been shown that our model can perform with high accuracy (less than 1% substitution error) while operating in the real-life environment to meet industrial standards. Considerable gain in recognition speed can be obtained by implementing this model on a special purpose hardware which supports parallel operations, and fast algorithm to retrieve shape information from the database.

References

ADAM-80 Adamo, J. M. (1980), "Fuzzy Decision Trees," Fuzzy Sets Syst., vol. 4, no. 3, pp. 207-220.

AGRA-77 Agrawala, A., K. and Kulkarni, A. V., (1977), "Sequential Approach to the Extraction of Shape Features," Computer Graphics and Image Processing, Vol. 6, 538-557.

AHME-85 Ahmed P., Siddiqui K. J. and Suen, C. Y., (1985), "A Dynamic Approach to Extract Shape Information and Knowledge Organization," Proc., IASTED International Symposium, pp. 249-252.

AHME-84 Ahmed, P. and Suen, C. Y., (1984), "Edge Classification and Extraction of Shape Features," 7th, Int. Conf. On Pattern Recognition, pp. 593-596.

AHME-82 Ahmed, P. and Suen, C. Y., (1982), "Segmentation of Unconstrained Handwritten Postal Zip Codes," Proc. Int. Conf. on Pattern Recognition, pp. 545-547.

ALI -77 Ali, F. and Pavlidis T. (1977a), "Description and Recognition of Handwritten Numerals," Proc. Workshop Picture Data Processing and Management, pp. 26-32.

ALI -77 Ali, F. and Pavlidis, T., (1977b), "Syntactic Recognition of Handwritten Numerals," IEEE Trans. System. Man Cybern., vol. SMC-7, pp. 537-541.

ANDR-71 Andrews, H. C., (1971), "Multi-dimensional Rotations in Feature Selection," IEEE Trans. Comput., vol. C-20, pp. 1045-1051.

ARCE-71 Arcelli, C. and Levialdi, S., (1971),"Concavity Extraction by Parallel Processing," IEEE, Trans. Systems Man Cybern., vol. SMC-1, pp. 349-396.

ARCE-73 Arcelli, C. and Levialdi, S., (1973), "On Blob Recognition,"Computer Graphics and Image Processing, Vol. 2, pp. 22-38.

BLAC-74 Blackwell, F. W. , (1974), "Combining Mathematical And Structural Pattern Recognition," Proc. 2nd Int. Joint Conf. Pattern Recognition, pp.

78-80.

BLED-59 Bledsoe, W. W. and Browning, I., (1959), "Pattern Recognition and Reading by Machine," Proc. East. Joint Computer Conf., pp. 225-232.

BRIB-80 Bribiesca, E. and Guzman, A., (1980), "How to Describe Pure Form and How to Measure Differences in Shapes Using Shape Numbers," Pattern Recognition vol. 12, pp. 101-112.

BULD-62 Buldreay, J. and Milbrandt, E., (1962), "Solving Registration Problems in OCR," Electronics, vol 5, pp. 77-78.

CALV-70 Calvert, T. W., (1970), "Nonorthogonal Projections for Feature Extraction in Pattern Recognition," IEEE Trans. Comput., vol. C-19, pp. 447-452.

CHAN-81 Chang, N., (1981), Image Analysis and Image Database Management, UMI Research Press, Ann Arbor, Michigan, pp. 42-61.

CHAN-77 Chang, R. L. P. and Pavlidis, T. (1977), "Fuzzy Decision Tree Algorithms," IEEE Trans. Syst. Man Cybern., SMC-7, pp. 28-34.

CHAT-62 Chattan, J. B. and Teacher, C. F., (1962), "Character Recognition Techniques for Address Reading," in Optical Character Recognition, (Eds., G. L. Fisher, D. K. Pollock, B. Raddak and M. E. Stevens) Spartan Book, New York, pp.51-59.

CHEN-65 Chen, C. H., (1965), "A Computer Searching Criterion for Best Feature Set in Character Recognition," Proc. IEEE, vol. 53, pp.2128-2129.

CHOW-57 Chow, C. K., (1957), "An Optimum Character Recognition System Using Decision Functions," IRE Trans. Elect. Comput., vol. 6, pp. 247-254.

CODD-70 Codd, E., F., (1970), "A Relational Model of Data for Large Shared Data Bank," CACM, vol. 13, no. 6, pp. 377-384.

COVE-67 Cover, T. M. and Hart, P. E. (1967), "Nearest Neighbour Pattern Classification," IEEE Trans. on Information Theory, vol. IT-13, no. 1, pp. 21-27.

DANI-78 Danielsson, P., (1978), "Note: A New Shape

Factor," Computer Graphics and Image Processing, Vol.7 pp. 292-299.

DATT-80 Dattatreya, G. R. and Sarma, V. V. S., (1980), "Decision Tree Design for Pattern Recognition Including Feature Measurement Cost," Proc. 5th Int. Conf. Pattern Recognition, pp. 1212-1214.

DAVI-77 Davis, L., (1977), "Understanding Shape:II Symmetry," IEEE Trans. on Syst., Man, Cybern SMC-7, no. 3, pp. 204-212.

DEIG-74 Deighton, P.D. and Bass A.E., (1974), "On Orthogonal Coding and Piecewise linear Discriminant Function in Alphanumeric Pattern Classification", Research Dept. Report No. 494, Project no. 80806000, Post Office Research, Dolls Hills, London, England.

DELU-72 Deluca, A. and Termini, S., (1972), "A Definition of a Nonprobabilistic Entropy in the Setting of Fuzzy Sets," Information and Control, vol. 20, pp. 301-312.

DELU-73 Deluca, A. and Capocelli, R. M., (1973), "Fuzzy Sets and Decision Theory," Information and Control, vol. 23, pp. 446-473.

DEPA-75 Depalma, G. F. and Yau, S. S., (1975), "Fractionally Fuzzy Grammars with Application to Pattern Recognition," in Fuzzy Sets and their Applications to Cognitive and Decision Processes (Eds. L. A. Zadeh, K. S. Fu, K. Tanaka, and M. Shimura), Academic Press, New York, pp. 329-351.

DOYL-60 Doyle, W., (1960), "Recognition of Sloppy, Hand-printed Characters," in Proc. West. Joint. Computer Conf., vol. 17, pp. 133-142.

DUDA-66 Duda, R. O. and Fossum, H., (1966), "Pattern Classification by Iteratively Determined Linear and Piecewise Linear Discriminant Functions," IEEE Trans. Electron. Comput., vol. C-15, pp. 220-232.

DUER-80 Duerr, B. et al, (1980), "A Combination of Statistical and Syntactical Pattern Recognition Applied to Classification of Unconstrained Handwritten Numerals," Pattern Recognition, vol. 12, pp. 189-199.

DUNN-74 Dunn, J. C. (1974), "A Graph Theoretic Analysis of Pattern Classification via Tamura's Fuzzy Relation," IEEE Trans. Syst. Man Cybern., vol. SMC-3, pp. 310-313.

FARI-83 Faris, B. and Behrouz, P., (1983), "Approximation of Multipath Planar Shapes in Pattern Recognition," Int. Journal Computer and Information Sciences, vol. 12, no. 2, pp. 99-110.

FOCH-76 Focht, L. R. and Burger, A., (1976), "A Numeric Script Recognition Process for Postal ZIP code Applications," Proc., IEEE. Int., Conf., Cybern., pp.489-492.

FREE-70 Freeman, H., (1970), "Boundary Encoding and Processing," in Picture Processing and Psychopictorics (Eds. B. S. Lipkin and A. Rosenfeld), Academic Press, New York, pp. 241-266.

FU -67 Fu, K. S., Chien, Y. T. and Gardillo, G. P., (1967), "A dynamic Programming Approach to Sequential Pattern Recognition," IEEE Trans. Comput., vol. C-16, pp. 790-803.

FU -74 Fu, K. S., (1974), Syntactic Methods in Pattern Recognition, Academic Press, New York and London.

GLUC-71 Glucksman, H. A., (1971), "Multicategory Classification of Pattern Represented by High-order Vectors of Multilevel Measurements," IEEE Trans. Comput., vol. C-20, pp. 1593-1598.

GONZ-77 Gonzalez, R. C. and Paul W.,(1977), Digital Image Processing, Addison-Wesley, Reading, Mass.

GRAN-72 Granlund, G. H., (1972), "Fourier Preprocessing for Hand Print Character Recognition," IEEE Trans. Comput., vol. C-21, pp. 195-201.

GUDE-76 Gudesen, A., (1976), "Quantitative Analysis of Preprocessing Techniques for the Recognition of Handprinted Characters," Pattern Recognition, vol. 8 pp. 219-227.

HENN-68 Hennis, R. B., (1968), "The IBM 1975 Optical Page Reader Part:I System Design," IBM Journal of Research and Development, vol. 12, no. 5, pp. 346-353.

HIGH-62 Highleyman, W. H., (1962), "Linear Decision Functions with Application to Pattern Recognition," Proc. IRE, 50, pp. 1501-1514.

HOFF-71 Hoffman, R. L. and McCullough, J. W., (1971), "Segmentation Methods for Recognition of Machine Printed Characters," IBM Journal of Research and Development, vol. 15, no.2, pp.189-199.

HUNT-72 Hunt, D. J., (1972), "A Feature Extraction Method for the Recognition of Handprinted Characters," in Machine Perception of Patterns and Pictures. London, England: The Institute of Physics, pp. 28-33.

HUSS-72 Hussain, A. B. S., Toussaint, G. T. and Donaldson, R. W., (1972), "Results Obtained using a simple character recognition procedure on Munson's handprinted data," IEEE Trans. Comput., vol.C-21, pp. 201-205.

JOHN-66 Johnson, D. D., et al (1966), "The Application of a Few Hyperplane Decision Techniques to Hand Printed Character Recognition," Proc. Nat. Electronic Conf., vol. 22, pp. 869-874.

KANA-72 Kanal, L. N. and Chandrasekran B., (1972), "On Linguistic, Statistical and Mixed Methods for Pattern Recognition," in Frontiers of Pattern Recognition, (Ed. Watanabe, S.), Academic Press, New York.

KATS-69 Katsuragi, S., et al (1969), "Recognition of Handwritten Numerals Using Decision Graph," Proc. 1st Int.Joint Conf. Artificial Intelligence, pp. 161-170.

KAUF-75 Kaufmann, A., (1975), Introduction To The Theory of Fuzzy Subsets vol-I, Academic Press, New York.

KICK-76 Kickert, W. J. M. and Koppelaar, H., (1976), "Application of Fuzzy Set Theory to Syntactic Pattern Recognition of Handwritten Capitals," IEEE Trans. Syst., Man, Cybern., vol. SMC-6, pp. 148-151.

KNOL-69 Knoll, A. L., (1969), "Experiments With 'Characteristic Loci' for Recognition of Handprinted Characters," IEEE Trans. Comput., vol.C-18 pp. 336-372.

KNUT-68 Knuth, D. E., (1968), "Semantics of Context-free

Languages," J. Math. Syst. Theory, vol 2, pp. 127-146.

KOOI-76 Kooi, R. and Lin, W. C., (1976), "An On-Line Minicomputer Based System for Reading Printed Text Aloud," Proc. IEEE Conf. Cybern. Soc., pp.509-513.

KRAM-73 Kramer, H. P., et al, (1973), "The Recognition of Handwritten Symbols and its Relation to Formal Languages," Proc. 1st Int. Joint Conf. Pattern Recognition, pp. 57-58.

KRIS-83 Krishnamoorthy, S. G., Ahmed, P. and Suen, C. Y. (1983) "Computer-aided Determination of Nearly Optimal Dot-matrix Size for Display of Indian Characters," in Proc. Of the Soc. For Information Display, Vol. 24, no. 3, pp. 271-279.

KUNI-74 Kunii, T., Weyl, S. and Tenenbaum, J. M., (1974), "A Relational Data Base Schema for Describibg Complex Pictures with Color and Texture," Proc. 2nd Int., Joint Conf. on Pattern Recognition, Copenhagen, Denmark, pp. 310-316.

LEE -80 Lee, J., (1980), "Digital Image Enhancement and Noise Filtering by Use of Local Statistics," IEEE Trans., on Pattern Analysis and Machine Intelligence, vol. PAMI-2, no. 2, pp. 165-168.

LEVI-69 Levine, M. D., (1969), "Feature Extraction: a Survey," Proc. IEEE, vol. 57, pp. 1391-1407.

LIN -80 Lin, Y. K. and Fu, K. S., (1980), "Automatic Classification of Cervical Cells Using a Binary Tree Classifier," Proc. 5th Int. Conf. Pattern Recognition, pp. 570-574.

MARI-60 Marill, T. and Green D. M., (1960), "Statistical Recognition Functions and the Design of Pattern Recognizer," IRE Trans. Elect. Computers, vol. 9, pp. 472-477.

MEIS-68 Meisel, W. S., (1968), "Least Square Methods in Abstract Pattern Recognition," Information Science, vol. 1, no. 1, pp. 43-54.

MINN-66 Minneman, M. J., (1966), "Handwritten Character Recognition Employing Topology, Cross-correlation, and Decision Theory," IEEE Trans. Syst. Sci. Cybern., vol. 2, pp. 86-96.

MINS-69 Minsky, M. and Papert, S., (1969), Perceptron, MIT Press, Cambridge.

MIVA-74 Mivazaki, T. and Hoshino, Y., (1974), "Letter Segmenting Device for OCR Comprising Multi-Level Segmentor Operable when Binary Segmentation Fails," Canadian Pat. No.1055160.

MORI-82 Mori, S. and Doh, M., (1982), "A Sequential Tracking Extraction of Shape Features and Its Constructive Description," Comput. Graphics and Image Processing, Vol. 19, no. 4, 349-366.

MUNS-68 Munson, J. H., (1968), "Experiments in the Recognition of Hand-printed Text: Part I - Character Recognition," in Proc. Fall Joint Computer Conf., vol.33, pp. 1125-1138.

NADL-74 Nadler, M., (1974), "Sequential-Local Operators, Proc. 2nd Int. Joint Conf. on Pattern Recognition, pp. 131-135.

NARA-69 Narasimhan, R., (1969), "On the Description, Generation, and Recognition of Classes of Pictures," in Automatic Interpretation and Classification of Images (Ed. A. Grasselli), Academic Press, New York, pp. 1-42.

NARA-71 Narasimhan, R. and Reddy, V. S. N., (1971), "A Syntax-aided Recognition Scheme for Hand Printed English Letters," Pattern Recognition, vol.3, pp. 345-361.

NEIL-69 Neill, J., (1969), "Numeric Script Mail Sorter," Proc. Automatic Pattern Recognition, pp. 49-65.

NILS-65 Nilsson, N. J., (1965), "Learning Machines-Foundations of Trainable Pattern-Classifying Systems," McGraw-Hill, New York and London.

PAL -81 Pal, S. K. and King, R. A., (1981), "Image Enhancement Using Smoothing with Fuzzy Sets," IEEE Trans. On Systems, Man, Cybern., vol SMC-11, no. 7, pp. 494-501.

PAL -83 Pal, S. K., King, R. A. and Hashim, A. A., (1983), "Automatic Grey Level Thresholding Through Index of Fuzziness and Entropy," Pattern Recognition Letters, vol. 1, pp. 141-146.

PAND-78 Panda, D. P., (1978), "Note: Nonlinear Smootihing of Pictures," Computer Graphics and Image Processing, vol. 8, pp. 259-170.

PARK-74 Parks, J. R. et al, (1974), "An articulate Recognition Procedure Applied to Handprinted Numerals," Proc. 2nd Int. Joint Conf. Pattern Recognition, pp. 416-420.

PERS-77 Persoon, E. and Fu, K. S., (1977), "Shape Discrimination Using Fourier Descriptors," IEEE Trans. Syst., Man, Cybern., SMC-7, no. 3, pp. 170-179.

PUN -80 Pun, T., (1980), "A New Method for Grey-level Picture Thresholding Using the Entropy of the Histogram," Signal Processing, vol. 2, pp. 223-237.

ROSE-76 Rosenfeld, A. and Kak, A. C., (1976), Digital Picture Processing, Academic Press, New York.

ROSE-66 Rosenfeld, A. and Pfaltz, J. L., (1966), "Sequential Operations in Digital Picture Processing," J. of the ACM, Vol.13, no.4, pp. 471-494.

ROSE-81 Rosenfeld, A. and Smith, R. C., (1981), "Thresholding Using Relaxation," IEEE Trans. On Pattern Analysis and Machine Intelligence, vol. PAMI-3, no. 5, pp. 598-606.

SCHA-82 Schantz, H. F., (1982), The History of OCR Optical Character Recognition, Recognition Technologies Users Association (RTUA), USA.

SHAW-69 Shaw, A. C., (1969), "A Formal Description Scheme as a Basis for Picture Processing System," Inform. & Contr., 14, pp.9-52.

SHEL-72 Shelman, C. B., (1972), "The Application of List Processing Techniques to Picture Processing," Pattern Recognition, Vol.4, pp.201-210.

SHIM-73 Shimura, M., (1973), "Multicategory Learning Classifiers for Character Reading," IEEE Trans. Syst., Man, Cybern., vol. SMC-3, pp.74-85.

SHRI-85 Shridhar, M. and Badreldin, A., (1985), "A High-Accuracy Syntactic Recognition Algorithm for Handwritten Numerals," IEEE Trans. on Systems, Man, and Cybern., vol. SMC-15, no. 1, pp. 152-158.

SIDD-84 Siddiqui, K. J. and Ahmed P.,(1984), Special Architecture for Optical Character Recognition and Image Processing, Technical Report Cat. No. HF5001C6+1984, No.46, Concordia University, Montreal.

SIY -74 Siy, P. and Chen, C. S., (1974), "Fuzzy Logic for Handwritten Numeral Character Recognition," IEEE Trans. Syst., Man, Cybern., vol. SMC-4, pp. 570-575.

SOLT-57 Soltran, (1957), "Typed Figures Translated into Computer Codes," Engineering, pp.348-349.

SPAN-76 Spanjersberg A. A., (1976), "Experiment With Automatic Input of Handwritten Numerical Data into a Large Administration System," Proc. Int. Conf. on Cybernetics and Soc., pp. 476-478.

SPAN-74 Spanjersberg, A. A., (1974), "Combinations of Different Systems for the Recognition of Handwritten Digits," Proc. 2nd Int. Joint Conf. Pattern Recognition, pp. 208-209.

SPEC-67 Specht, D. F., (1967), "Generation of Polynomial Discriminant Functions for Pattern Recognition," IEEE Trans. Electron. Comput., vol. 16, no. 3, pp. 308-319.

STAL-77 Stallings, W., (1977), "Fuzzy Set Theory Versus Basian Statistics," IEEE Trans. Syst., Man, Cybern., vol SMC-7, pp. 216-219.

SUEN-86 Suen, C. Y., (1986), "Character Recognition by Computer and Applications," in T. Y. Young and K. S. Fu (eds.), Handbook of Pattern Recognition and Image Processing, Academic Press, New York.

SUEN-82 Suen, C. Y., (1982a), "The Role of Multi-Directional LOCI and Clustering in Reliable Recognition of Characters," Proc. 6th Int. Conf. on Patern Recognition, pp. 1020-1022.

SUEN-82 Suen, C. Y., (1982b), "Distinctive Features in Automatic Recognition of Handprinted Characters," Signal Processing, vol. 4, pp. 193-207.

SUEN-80 Suen, C. Y., Berthod, M. and Mori, S., (1980), "Automatic Recognition of Hanprinted Characters," Proc. IEEE, 68, no. 4, 469-487.

SUEN-74 Suen, C. Y., (1974), "Human Factors in Character Recognition," Proc. Int. Conf. Cybern. Soc., pp.253-258.

SUEN-73 Suen, C. Y., (1973), "Factors Affecting the Recognition of Handprinted Characters," Proc. Int. Conf. Cybern. Soc., pp.174-175.

SWAI-77 Swain, P. H. and Hauska, H. (1977), "The Decision Tree Classifier: Design and Potential," IEEE Trans. Geosci. Electron., vol. GE-15, pp. 142-147.

TAMU-71 Tamura, S., Higuchi, S. and Tanaka, K., (1971), "Pattern Classification Based on Fuzzy Relations, IEEE Trans. Syst., Man, Cybern., vol. SMC-1, pp. 61-66.

THOM-75 Thomason, M. G., (1975), "Finite Fuzzy Automata, Regular Languages and Pattern Recognition," Pattern Recognition, vol. 5, pp. 383-390.

TIEN-85 Tien Dung La, (1985), "A Cluster-Based Classification Method for the Recognition of Unconstrained Handwritten Numerals," Tech. Report, Dept. of Computer Science, Concordia University, Montreal, Canada.

TOU -72 Tou J. T. and Gonzalez, R. C., (1972), "Recognition of Handwritten Characters by Topological Feature Extraction and Multilevel Categorization," IEEE Trans. Comput., vol.C-21, pp. 776-785.

TSAI-80 Tsai, W. and Fu, K. S., (1980), "Attributed Grammar-A tool for Combining Syntactic and Statistical Approaches to Pattern Recognition," IEEE, Trans. on Systems, Man Cybern., vol. SMC-10, no. 12, pp. 873-885.

ULLM-73 Ullmann, J. R., (1973), Pattern Recognition Techniques, Crane Russak and Company Inc., New York.

ULLM-74 Ullmann, J. R., (1974), "Automatic Selection of Reference Data for Use in Nearest Neighbor Method of Pattern Classification," IEEE Trans. Information Theory, vol. 20, pp. 541-543.

UNDE-70 Underwood, E. E., (1970), Quantitative Stereology, Addison-Wesley, Reading, Mass.

WANG-84 Wang, Q. R. and Suen, C. Y., (1984), "Analysis and Design of a Decision Tree Based on Entropy Reduction and its Application to Large Character Set Recognition," IEEE, Trans. on Pattern Recog. and Machine Intel., vol PAMI-6, no. 4, pp. 406-417.

WATT-71 Watt, A. H, and Beurle, R. L., (1971), "Recognition of Handprinted Numerals Reduced to Graph-representable Form," Proc. 2nd Int. Joint Conf. Artificial Intelligence, pp. 322-332.

WEE -69 Wee, W. G. and Fu, K. S., (1969), "A Formulation of Fuzzy Automata and its Application as a Model of Learning Systems," IEEE Trans. Syst., Man, Cybern., vol 5, pp. 215-223.

WEND-76 Wendling, S. and Stamon, G, (1976),"Hadamard and Harr Transforms and Their Power Spectra in Character Recognition," in Proc. Joint Workshop Pattern Recognition and Artificial Intelligence, pp. 103-112.

WESZ-78 Weszka, J. S., (1978), "A Survey of Threshold Selection Techniques," Computer Graphics and Image Processing, vol. 7, pp. 259-268.

WHIT-83 White, J. M. and Rohrer, G. D., (1983), "Image Thresholding for Optical Character Recognition and Other Applications Requiring Character Image Extraction," IBM Journal of Research and Development, vol. 27, no. 4, pp. 400-410.

WILL-73 William S. M. and Demetrios A. M., (1973), "A Partitioning Algorithm with Application in Pattern Classification and the Optimization of Decision Trees," IEEE Trans. Comput., vol. C-22, no. 1, pp. 93-103.

ZADE-65 Zadeh, L. A. (1965), "Fuzzy Sets," Information and Control, vol. 8, pp. 338-353.

ZAHN-72 Zahn, C. and Roskies, R., (1972), "Fourier Descriptors for Palane Closed Curves,"IEEE Trans. Comput., vol. C-21, pp. 269-280.

ZHAN-80 Zhang S., (1980), "The Regular Expressions Inference for Syntactic Recognition of Handwritten Numerals," Proc. 5th Int. Conf. on Pattern Recognition, pp. 1004-1006.

Appendix A

Confusion Tables

Table Number 1 (a)
Prediction Module
Training Set Size 5000
Unique Choice

	0	1	2	3	4	5	6	7	8	9	Samples Processed	Samples Rejected	Samples Substituted	Samples Per class
0	184	0	0	0	0	0	0	0	0	0	184	316	0	500
1	0	375	0	0	0	0	0	0	0	0	375	125	0	500
2	0	0	78	3	0	0	0	1	0	0	82	418	4	500
3	0	0	0	175	0	0	0	0	1	0	176	324	1	500
4	0	0	0	0	121	0	0	0	0	7	128	372	7	500
5	0	0	0	6	0	139	0	0	0	3	148	352	9	500
6	0	0	0	0	0	0	145	0	0	0	145	355	0	500
7	1	0	0	0	0	0	0	172	0	0	173	327	1	500
8	0	0	0	6	0	0	0	0	136	0	142	358	6	500
9	0	0	0	0	0	0	0	0	0	173	173	327	0	500
	185	375	78	190	121	139	145	173	137	183	1726	3274	28	5000

Recognition Reliability = 98.38%
Substitution Rate = .56%
Rejection Rate = 65.48%

Table Number 1 (b)
Prediction Module
Testing Set Size 3540
Unique Choice

	0	1	2	3	4	5	6	7	8	9	Samples Processed	Samples Rejected	Samples Substituted	Samples Per class
0	269	0	0	0	0	0	0	0	1	0	270	390	1	660
1	0	475	0	0	0	0	0	0	1	0	476	184	1	660
2	0	0	15	0	0	0	0	1	0	0	16	144	1	160
3	0	0	0	44	0	0	0	0	1	0	45	115	1	160
4	0	0	0	0	87	0	0	0	0	5	92	228	5	320
5	0	0	0	0	0	20	0	0	0	0	20	80	0	100
6	0	0	0	0	0	1	160	0	0	0	161	439	1	600
7	1	0	0	0	1	0	0	190	0	0	192	408	2	600
8	0	1	0	2	0	0	0	0	28	0	31	109	3	140
9	0	0	0	0	0	0	0	0	1	45	46	94	1	140
	270	476	15	46	88	21	160	191	32	50	1349	2191	16	3540

Recognition Reliability = 98.81%
Substitution Rate = .45%
Rejection Rate = 61.89%

Table Number 2 (b)
Prediction Module
Testing Set Size 3540
Two Choices

	0	1	2	3	4	5	6	7	8	9	Samples Processed	Samples Rejected	Samples Substituted	Samples Per class
0	209	1	0	1	0	0	0	0	3	0	214	176	5	390
1	0	102	4	0	0	0	0	1	4	0	111	73	9	184
2	0	0	42	6	0	0	0	1	1	0	50	94	8	144
3	1	0	0	62	0	1	0	1	3	1	69	46	7	115
4	2	2	0	0	99	0	1	0	0	20	124	104	25	228
5	0	0	0	4	0	31	2	0	3	1	41	39	10	80
6	2	2	0	2	1	1	180	0	1	0	189	250	9	439
7	4	7	0	0	2	0	0	180	1	2	196	212	16	408
8	0	0	0	5	1	0	0	0	33	1	40	69	7	109
9	0	1	0	0	1	0	0	0	0	52	54	40	2	94
	218	115	46	80	104	33	183	183	49	77	1088	1103	98	2191

Recognition Reliability = 90.99%
Substitution Rate = 4.47%
Rejection Rate = 50.34%

Table Number 2 (a)
Prediction Module
Training Set Size 5000
Two Choices

	0	1	2	3	4	5	6	7	8	9	Samples Processed	Samples Rejected	Samples Substituted	Samples Per class
0	138	0	0	0	0	0	0	0	1	0	139	177	1	316
1	0	71	2	0	0	0	0	0	3	0	76	49	5	125
2	1	1	114	10	0	2	0	5	4	0	137	281	23	418
3	0	0	0	183	0	0	0	0	2	0	185	139	2	324
4	2	3	0	0	178	0	0	0	0	26	209	163	31	372
5	1	0	0	34	3	124	1	0	2	3	168	184	44	352
6	2	2	0	0	0	7	164	0	1	0	176	179	12	355
7	1	6	0	0	1	0	0	159	2	2	171	156	12	327
8	0	3	0	12	0	1	0	0	166	6	188	170	22	358
9	1	3	0	2	4	0	0	4	1	154	169	158	15	327
	146	89	116	241	186	134	165	168	182	191	1618	1656	167	3274

Recognition Reliability = 89.68%
Substitution Rate = 5.10%
Rejection Rate = 50.58%

Table Number 3 (a)
Prediction Module
Training Set Size 5000
Three Choices

	0	1	2	3	4	5	6	7	8	9	Samples Processed	Samples Rejected	Samples Substituted	Samples Per class
0	84	2	0	0	0	1	0	0	3	0	90	87	6	177
1	0	19	2	1	0	0	0	0	3	0	25	24	6	49
2	6	2	92	10	0	3	1	5	7	1	127	154	35	281
3	2	2	1	87	0	2	0	2	5	1	102	37	15	139
4	1	7	0	0	73	0	1	0	3	11	96	67	23	163
5	5	3	0	23	3	70	1	0	5	5	115	68	45	184
6	2	5	0	0	0	3	107	0	5	0	122	57	15	179
7	1	3	0	0	3	1	0	85	2	2	97	59	12	156
8	0	2	1	9	0	1	0	0	76	6	95	75	19	170
9	0	1	0	2	0	0	0	2	5	74	84	74	10	158
	101	46	96	132	79	81	110	94	114	100	953	703	186	1656

Recognition Reliability = 80.48%
Substitution Rate = 11.23%
Rejection Rate = 42.45%

Table Number 3 (b)
Prediction Module
Testing Set Size 3540
Three Choices

	0	1	2	3	4	5	6	7	8	9	Samples Processed	Samples Rejected	Samples Substituted	Samples Per class
0	115	0	1	0	0	3	2	0	3	0	124	52	9	176
1	1	24	6	1	0	0	0	2	3	1	38	35	14	73
2	1	0	31	6	0	0	1	1	3	0	43	51	12	94
3	0	0	0	26	0	1	1	1	5	0	34	12	8	46
4	6	2	0	1	38	0	1	1	2	12	63	41	25	104
5	0	0	0	1	0	22	0	0	1	3	27	12	5	39
6	15	4	0	5	0	6	108	0	7	0	145	105	37	250
7	5	6	0	3	1	1	0	106	4	7	133	79	27	212
8	0	0	1	2	0	0	0	0	23	5	31	38	8	69
9	0	1	1	0	2	0	0	0	2	15	21	19	6	40
	143	37	40	45	41	33	113	111	53	43	659	444	151	1103

Recognition Reliability = 77.09%
Substitution Rate = 13.69%
Rejection Rate = 40.25%

Table Number 4 (a)
Prediction Module
Training Set Size 5000
Four Choices

	0	1	2	3	4	5	6	7	8	9	Samples Processed	Samples Rejected	Samples Substituted	Samples Per class
0	48	1	0	0	0	1	1	1	0	0	52	35	4	87
1	0	5	1	3	0	0	0	1	1	0	11	13	6	24
2	4	2	46	7	0	2	3	4	7	0	75	79	29	154
3	0	2	1	13	0	2	0	0	1	1	20	17	7	37
4	2	1	0	0	18	1	1	1	1	8	33	34	15	67
5	2	0	0	6	1	17	1	0	0	1	28	41	11	69
6	5	1	0	0	0	2	29	0	0	0	37	20	8	57
7	2	0	0	0	1	0	0	28	1	2	34	25	6	59
8	0	2	1	5	0	2	0	2	25	6	43	32	18	75
9	1	1	0	1	4	0	1	0	4	24	36	38	12	74
	64	15	49	35	24	27	36	37	40	42	369	334	116	703

Recognition Reliability = 68.56%
Substitution Rate = 16.50%
Rejection Rate = 47.51%

Table Number 4 (b)
Prediction Module
Testing Set Size 3540
Four Choices

	0	1	2	3	4	5	6	7	8	9	Samples Processed	Samples Rejected	Samples Substituted	Samples Per class
0	23	1	0	1	0	0	2	0	4	0	31	21	8	52
1	0	7	1	2	0	0	0	1	2	0	13	22	6	35
2	4	1	9	3	0	2	1	5	2	0	27	24	18	51
3	0	0	0	5	0	1	0	0	0	0	6	6	1	12
4	4	1	0	0	8	1	1	0	3	5	23	18	15	41
5	0	0	0	1	0	6	0	0	1	0	8	4	2	12
6	8	1	1	4	2	3	29	0	4	1	53	52	24	105
7	4	2	0	0	4	0	0	26	0	1	37	42	11	79
8	0	0	0	3	0	1	0	1	9	2	16	22	7	38
9	0	2	0	0	1	0	0	2	0	3	8	11	5	19
	43	15	11	19	15	14	33	35	25	12	222	222	97	444

Recognition Reliability = 56.31%
Substitution Rate = 21.85%
Rejection Rate = 50.00%

Table Number 5 (a)
Prediction Module
Training Set Size 5000
Five Choices

	0	1	2	3	4	5	6	7	8	9	Samples Processed	Samples Rejected	Samples Substituted	Samples Per class
0	15	0	0	0	0	0	0	0	2	0	17	18	2	35
1	0	2	0	0	0	0	0	0	1	0	3	10	1	13
2	0	0	12	2	0	0	1	0	1	0	16	63	4	79
3	0	0	0	2	0	0	0	0	1	1	4	13	2	17
4	0	1	0	0	4	0	0	0	0	1	6	28	2	34
5	0	0	0	1	0	1	1	0	0	1	4	37	3	41
6	0	0	0	0	0	0	3	0	0	0	3	17	0	20
7	0	0	0	0	0	0	0	1	0	1	2	23	1	25
8	0	0	0	3	1	0	0	0	4	0	8	24	4	32
9	0	0	1	0	0	0	0	1	2	1	5	33	4	38
	15	3	13	8	5	1	5	2	11	5	68	266	23	334

Recognition Reliability = 66.18%
Substitution Rate = 6.89%
Rejection Rate = 79.64%

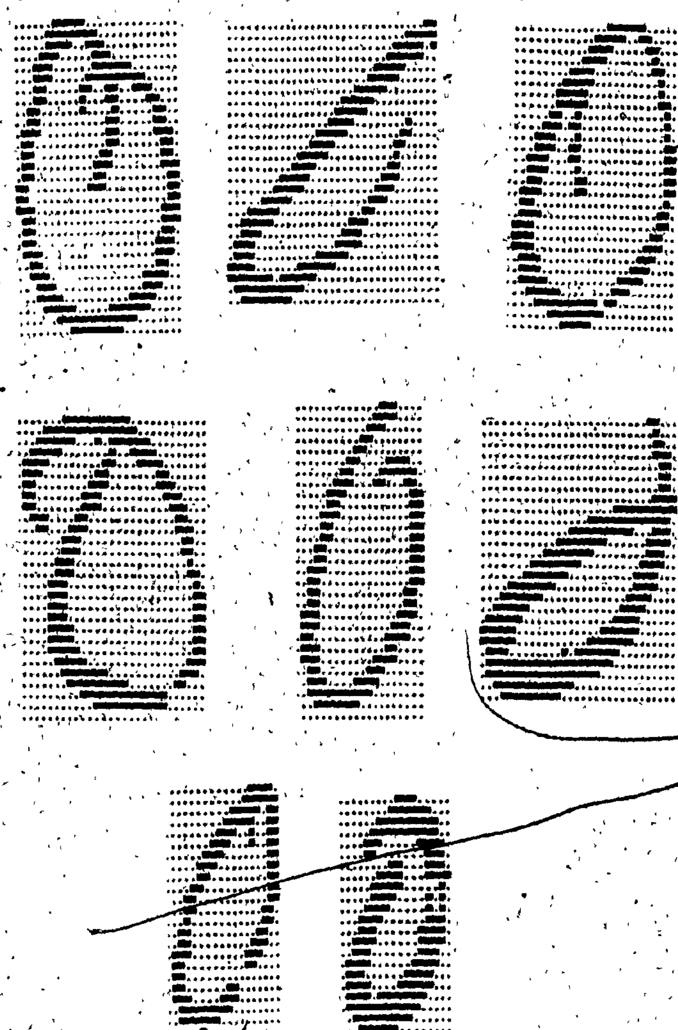
Table Number 5 (b)
Prediction Module
Testing Set Size 3540
Five Choices

	0	1	2	3	4	5	6	7	8	9	Samples Processed	Samples Rejected	Samples Substituted	Samples Per class
0	1	0	0	0	1	0	0	0	1	0	3	18	2	21
1	0	3	0	1	0	0	0	0	0	0	4	18	1	22
2	0	0	2	1	1	0	0	0	1	0	5	19	3	24
3	0	0	0	0	0	0	0	0	1	0	1	5	1	6
4	0	0	0	0	2	0	0	0	0	1	3	15	1	18
5	0	0	0	0	0	0	0	0	0	0	0	4	0	4
6	2	0	0	1	1	1	3	0	0	0	8	44	5	52
7	3	0	0	0	0	0	0	2	3	0	8	34	6	42
8	1	1	0	0	0	0	0	0	1	0	3	19	2	22
9	0	0	0	0	0	0	0	1	0	1	2	9	1	11
	7	4	2	3	5	1	3	3	7	2	37	185	22	222

Recognition Reliability = 40.54%
Substitution Rate = 9.91%
Rejection Rate = 83.33%

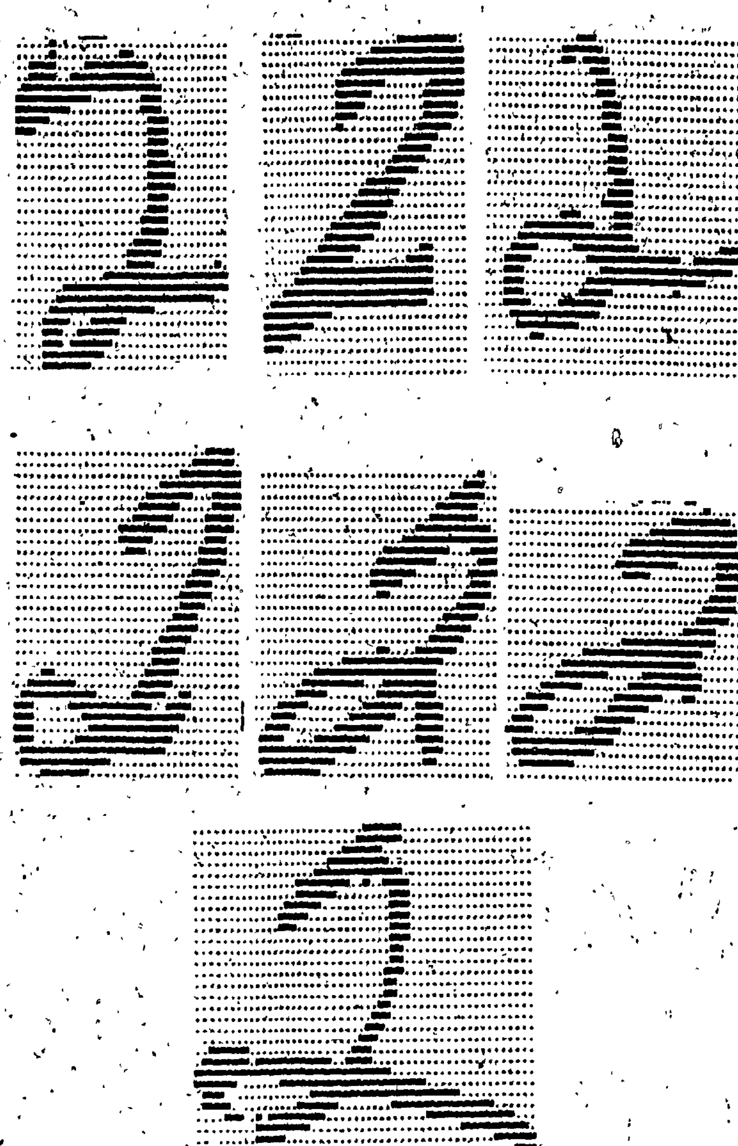
Appendix B

Samples From The Data Set.

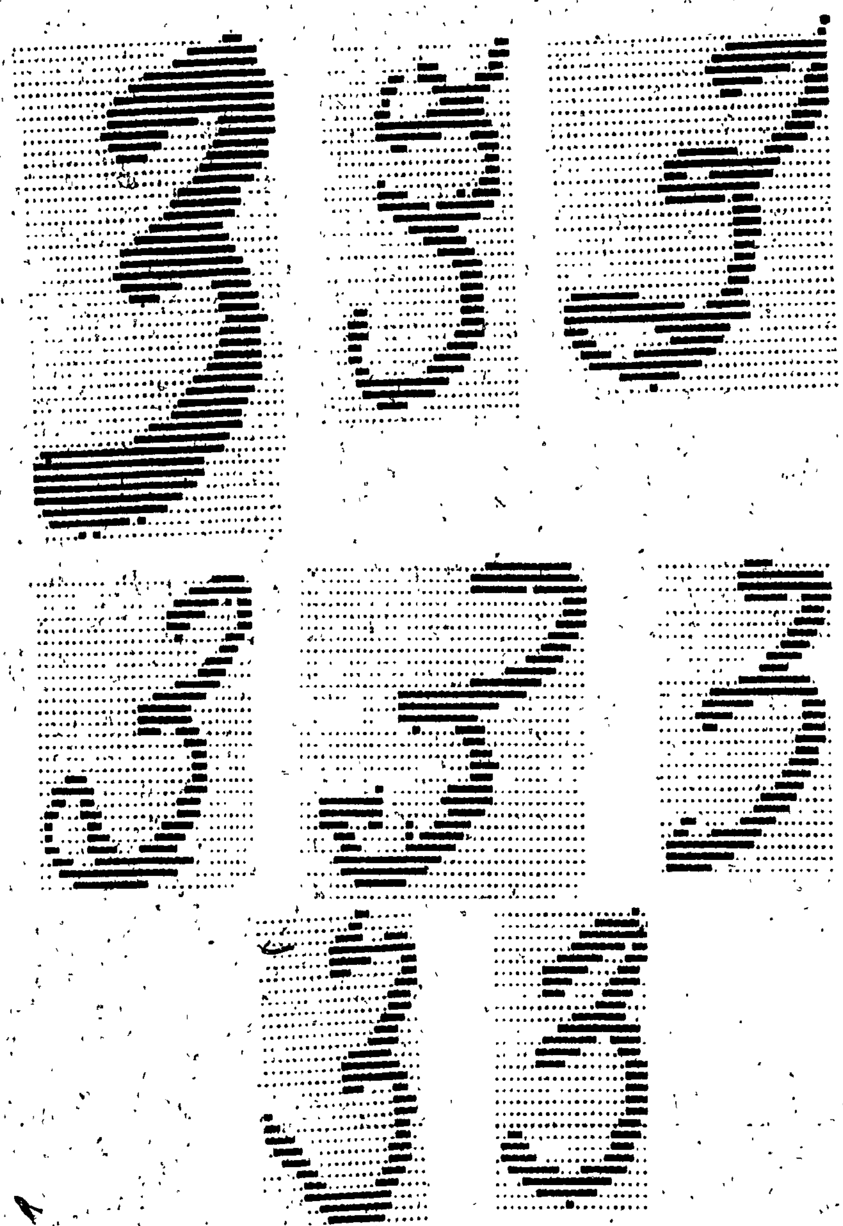


Samples of Zero.

Samples of One.

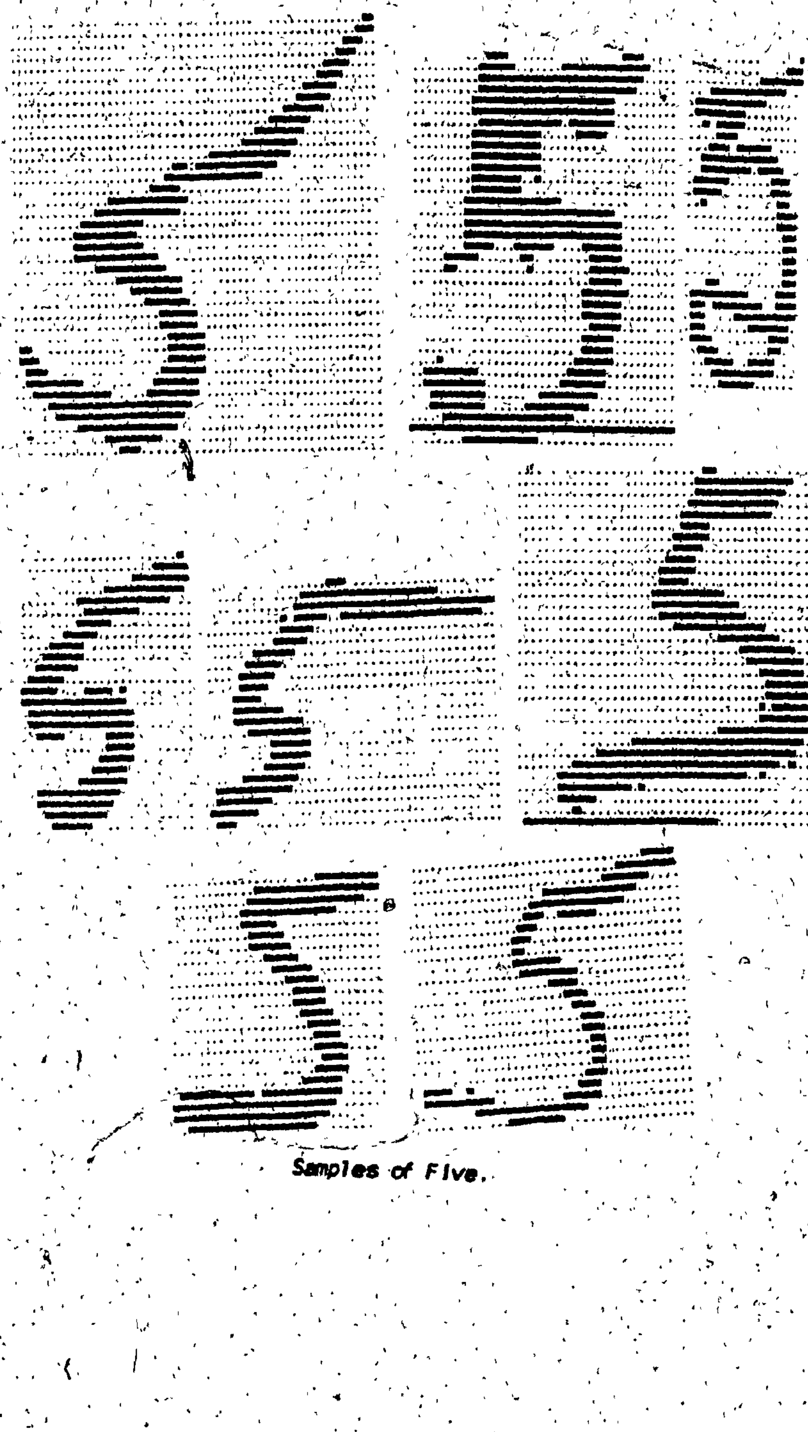


Samples of Two.

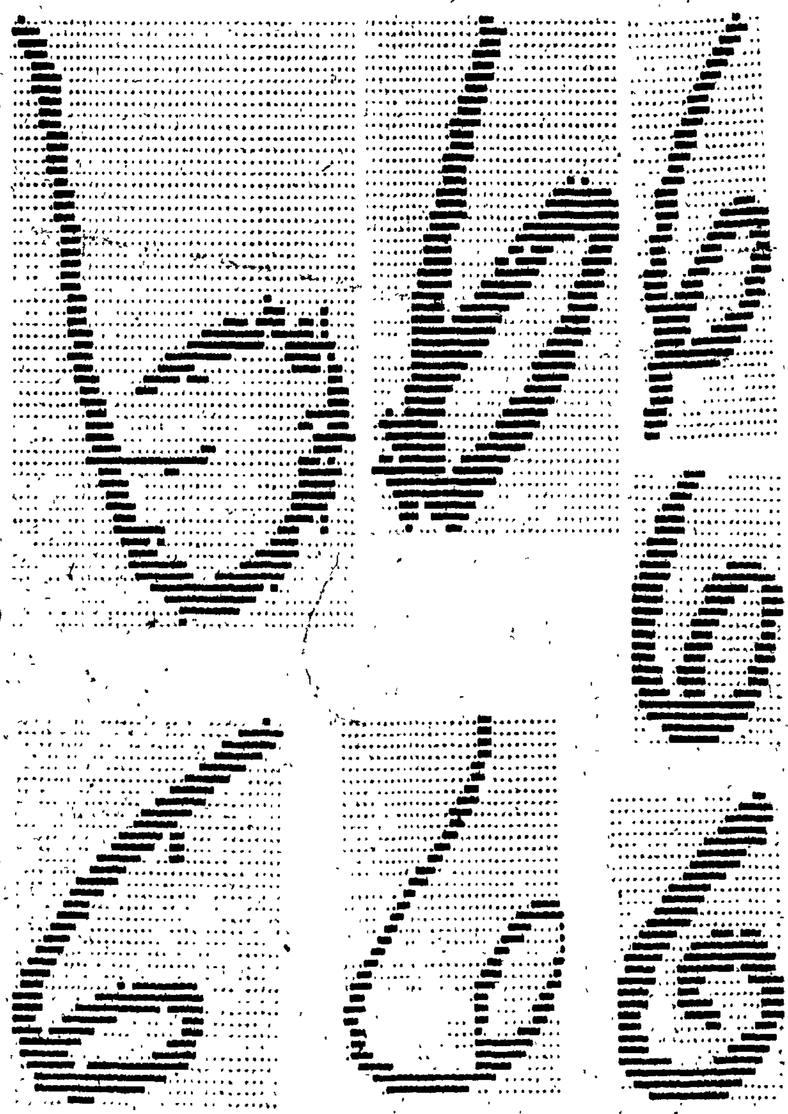


Samples of Three.

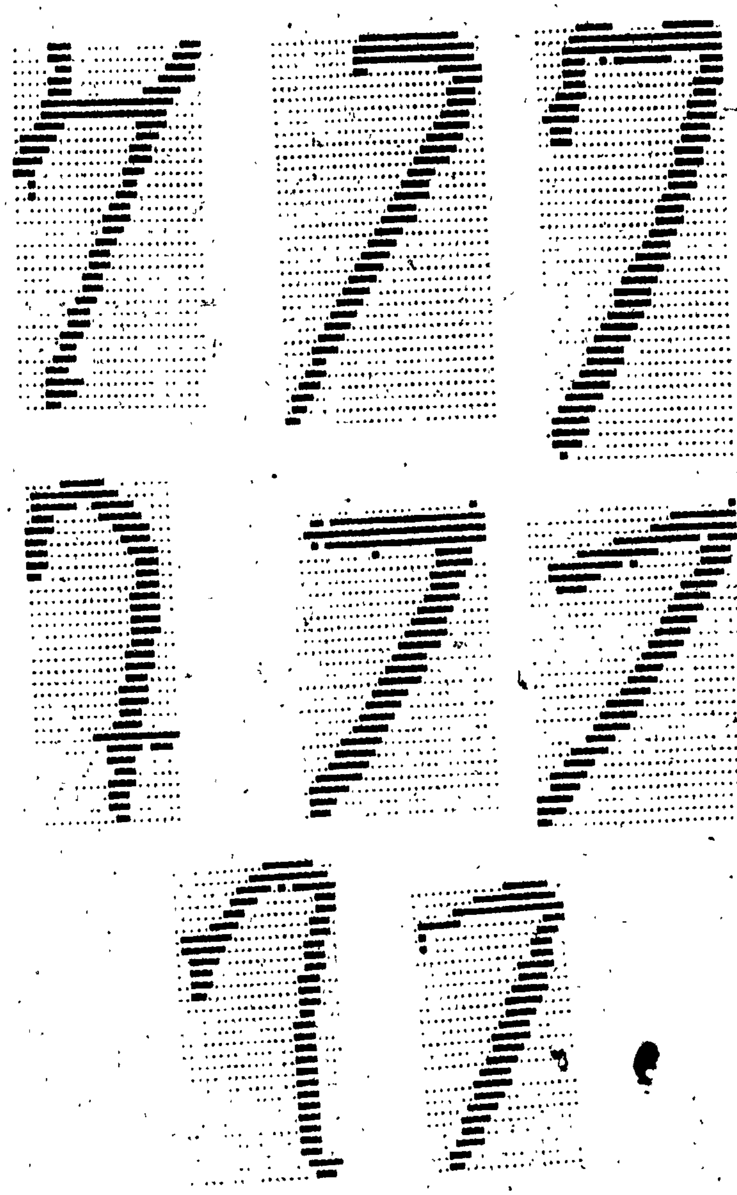
Samples of Four.



Samples of Five.



Samples of Six.



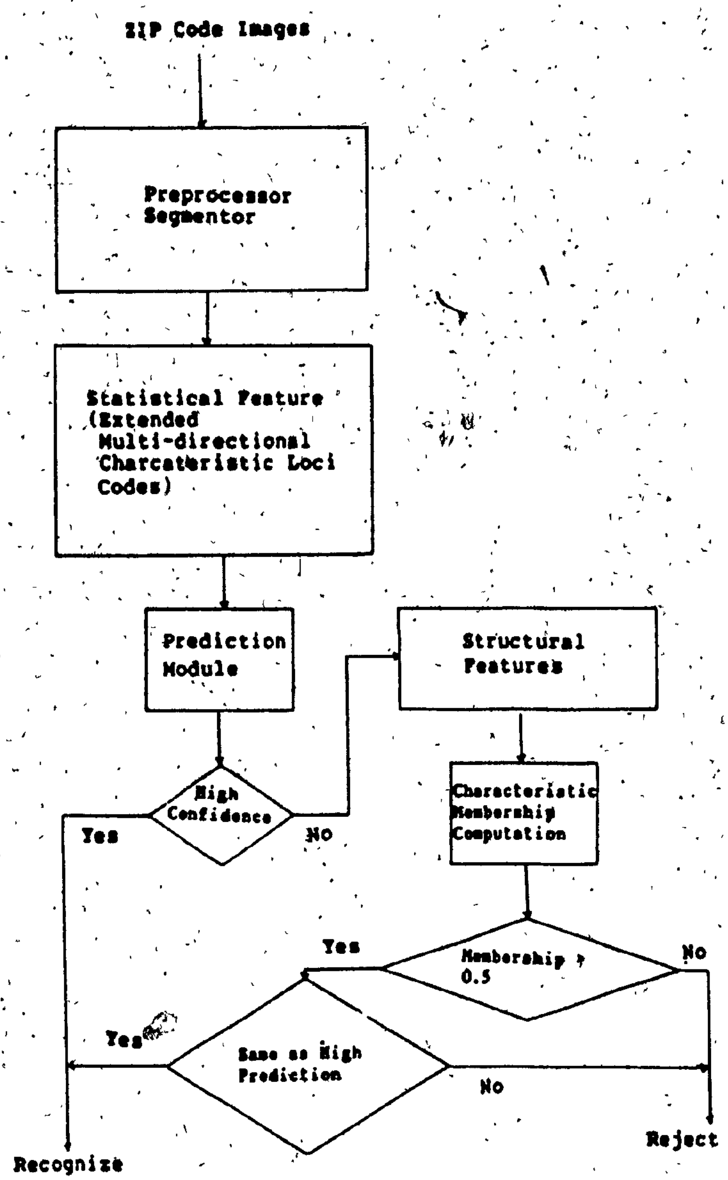
Samples of Seven.

Samples of Eight.

Samples of Nine.

Appendix C

Recognition Process.



A Flowchart Depicting The Complete Recognition Process.