National Library of Canada

Bibliothèque nationale du Canada

Acquisitions and Bibliographic Services Branch

Direction des acquisitions et des services bibliographiques

395 Wellington Street
Ottawa, Ontario
K1A 0N4

395, rue Wellington
Ottawa (Ontario)
K1A 0N4

NOTICE

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

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

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

Reproduction in full or in part of this microform is governed by the Canadian Copyright Act, R.S.C. 1970, c. C-30, and subsequent amendments.

AVIS

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

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

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

La reproduction, même partielle, de cette microforme est soumise à la Loi canadienne sur le droit d'auteur, SRC 1970, c. C-30, et ses amendements subséquents.

Canada

A Multi-Expert System
for Classifying
Unconstrained Handwritten Numerals

Sam Adel

A Thesis

in

The Department

of

Computer Science

Presented in Partial Fulfilment of the Requirements for
the Degree of Master of Computer Science at
Concordia University
Montreal, Quebec, Canada

March 1993

© Sam Adel, 1993

National Library of Canada
Bibliothèque nationale du Canada

Acquisitions and Bibliographic Services Branch
Direction des acquisitions et des services bibliographiques

395 Wellington Street
Ottawa, Ontario
K1A 0N4

395, rue Wellington
Ottawa (Ontario)
K1A 0N4

Your file Votre référence

Our file Notre référence

The author has granted an irrevocable non-exclusive licence allowing the National Library of Canada to reproduce, loan, distribute or sell copies of his/her thesis by any means and in any form or format, making this thesis available to interested persons.

L'auteur a accordé une licence irrévocable et non exclusive permettant à la Bibliothèque nationale du Canada de reproduire, prêter, distribuer ou vendre des copies de sa thèse de quelque manière et sous quelque forme que ce soit pour mettre des exemplaires de cette thèse à la disposition des personnes intéressées.

The author retains ownership of the copyright in his/her thesis. Neither the thesis nor substantial extracts from it may be printed or otherwise reproduced without his/her permission.

L'auteur conserve la propriété du droit d'auteur qui protège sa thèse. Ni la thèse ni des extraits substantiels de celle-ci ne doivent être imprimés ou autrement reproduits sans son autorisation.

ISBN 0-315-87296-9

Canada

Abstract

A Multi-Expert System
for Classifying
Unconstrained Handwritten Numerals

Sam Adel

There are infinite styles of handwriting and humans have no problem reading and recognizing them Until now, computers using mathematical models have failed to match human performance in classifying handwritten characters Classification of handwritten characters becomes especially difficult when computers are faced with confusing characters.

This thesis describes and analyses the design, implementation and testing of a new character recognition system for the recognition of unconstrained handwritten numerals. The system models and uses multiple human expertise in this field and focuses on the recognition of confusing cases of handwritten numerals

Test results of this numeral recognition system indicate that the system maintained low substitution rates with different data sets and consistently improved on its recognition rate by further training. Also that, in spite of the limitations of its knowledge base (in terms of scope and quality), its performance is comparable to that of some complete recognition systems which do not use human expertise. In the 'Conclusion' section of this thesis there are suggestions for improving knowledge acquisition and implementation of the system The suggested improvements should enhance the system significantly so that its performance will approach that of humans'.

This system differs from other recognition systems in the way it handles and uses human knowledge in this field and its original method of defining and grouping subclasses of numerals. These alone or the system as a whole might have an effect on the future direction in this area

Acknowledgements

I would like to take this opportunity to express my sincerest thanks to my thesis supervisor, Professor C. Y. Suen, for guidance, invaluable suggestions, criticisms and encouragement throughout the course of this research work.

I would like to thank Dr L. Lam and Mr. R. Legault for their suggestions and help

I am also grateful for the financial assistance provided by research grants awarded to Professor C. Y. Suen by the Natural Sciences and Engineering Research Council of Canada, the Ministry of Education of Quebec and the National Networks of Centres of Excellence Research Program of Canada.

Contents

List Of Figures

List Of Tables

Chapter 1

Introduction

For more than three decades hundreds of researchers, in many countries, have attempted to use computers to develop handwritten character recognition systems that can match human performance in this field ([3], [4], [5], [11], [12] and [16])

While the on-line character recognition systems are becoming commercially available, the off-line ones still remain as a challenge for researchers ([5] and [35])

Inability of ad hoc mathematical models to recognize handwritten characters (well) has been diverting the attention of researchers towards other avenues Use of models that are similar to the way that humans recognize characters has proven to be very attractive ([18] and [20]).

The problem of recognizing unconstrained handwritten numerals is in fact due to the confusing cases of handwritten numerals ([13] and [27]). Hence this work concentrates on the real problem, i.e., recognition of confusing cases of handwritten numerals ([9]).

LeClair ([34]), believes that expert system developers who consider only a single expert's heuristics and reasoning risk becoming "myopic". This is true at least in case of those domains of human knowledge that are not defined by concrete formulas or rules, like medicine or character recognition. Experts in these domains may have their own unique method of approach for providing answers to questions. Possibly better answers must be found if a synthesis of expertise can be created through a meaningful interaction among experts

In this work, a Multi-expert System ([7] and [15]) is used to model the expertise of five human experts in identifying confusing cases of numerals However, in order to get a better understanding of the design, implementation and performance issues of such a system, the construction of a complete system (involving simple and confusing cases) is discussed

1.1 Pattern Recognition in Man and Machine

The process of recognition and classification is one of the most fundamental of human activities. As a matter of fact, one of the most primitive and common activities of animals consists of sorting similar items into groups ([25]).

"A picture is worth a thousand words." or more accurately, a simple image can contain more than 25 million bits of information ([10]) and yet visual perception feels to us like a trivial task

Visual pattern recognition and classification by computers has evolved over many years from the use of computers for the recognition of maps and characters to the recognition of sophisticated scenes. In spite of limited success in machine vision, it is still very far from human visual capability

The main drawback of the existing computer recognition systems is the lack of extensibility of their techniques, i.e., algorithms that work fine in a simple domain often fail when the domain is extended. In contrast, the human visual system rapidly and effortlessly accomplishes different recognition tasks. Therefore it must be possible to enhance the performance of the man made recognition systems by studying and modeling the superior human recognition system ([10] and [20]).

1.2 Recognition of Confusing Cases of Unconstrained Handwritten Numerals

It is trivial to mention that; in a character recognition system the main cause of substitutions and rejections is due to samples which exhibit features belonging to more than one class of numerals, see Figure 1.1.

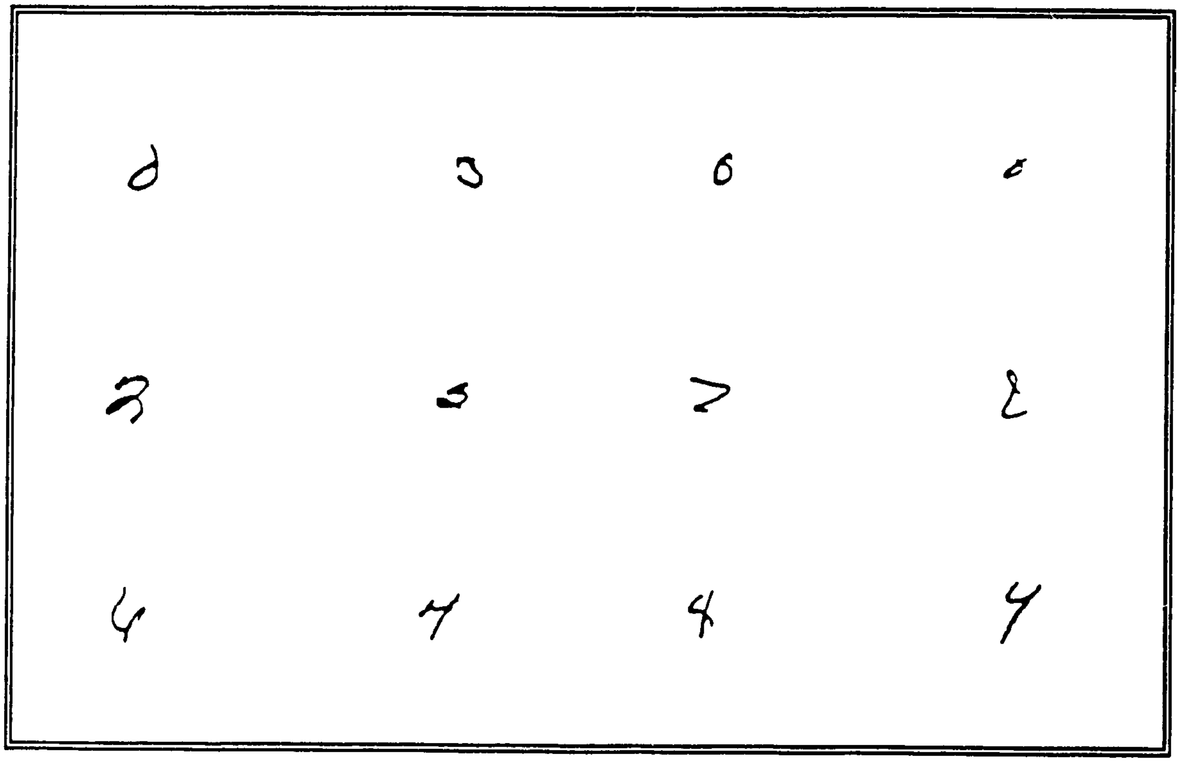


Figure 1.1: Examples of confusing cases of handwritten numerals.

Unfortunately this trivial fact has not been given enough attention by researchers ([9] and [13]). Most of the numeral recognition systems that have been developed so far identify only pairs or triplets of confusing classes of numerals, for example the confusing pair (1, 7) A more detailed study of the confusing cases of numerals might lead us to a specific method for their recognition ([13] and [28]).

One important finding of this work is that; each class of numerals has its own set of subclasses and subclasses of different classes form specific groups of subclasses. Subclasses within each group share many features, thereby causing confusion to the recognition systems. Analysing the comments of five human experts (regarding the identification of confusing cases of numerals) lead to the formation of subclasses and their groups.

By focusing on subclasses of numerals rather than classes of numerals, it should be possible to achieve high recognition and low substitution rates with different sets of data.

1.3 Multi-expert System for Modeling Human Recognition of Numerals

There is no specific formula for the recognition of unconstrained handwritten numerals. Recognition of numerals is another of those activities that humans do without any problem, using rule-of-thumb ([44]).

Expert Systems have proven to be successful in mimieking human expertise in those domains which deal with inexactness, e.g., medicine and law ([7]) Expert Systems have been used in pattern recognition as well ([14]), including character recognition ([2], [5] and [39]).

Getting a second (or third) opinion is another of those common things that humans do. Using multiple expertise in expert systems has been always an attractive option. Though, this bears the problem of aggregating the different opinions of experts.

Human knowledge in Expert Systems is modelled using (basically) production rules When these rules can not be defined precisely, it becomes almost impossible to use precise terms (logic) to define them. Fuzzy logic was formulated ([24] and [25]) as an alternative to such problems. That is when vague or uncertain human knowledge can be represented directly instead of using precise formulas ([32]) Fuzzy Logic has been found to be applicable in many areas, namely; business, psychology, engineering, ... and expert systems ([43])

Fuzzy Logic is an ideal tool for representing (acquiring) that part of human knowledge which is imprecise and Expert Systems are ideal tools for storing (organizing) and processing that imprecise knowledge Hence their combination must provide a powerful tool for replicating human expertise in certain domains such as recognition of unconstrained handwritten characters ([25])

1.4 The Proposed Numeral Recognition System

The major goal of this thesis is to examine the development process and performance of a recognition system that classifies confusing cases of unconstrained handwritten numerals using multiple human expertise in this area. Though in order to be able to analyse and evaluate this system more rigorously, a complete system (covering both confusing and simple cases of numerals) is implemented. The basic components of this system are as follows:

1. Preprocessor; an input character is binarized and skeletonized.
2. Feature Extractor, the skeleton of an input character is traced and segmented into primitives (lines, curves, cavities and loops). Features are extracted by analysing and synthesizing these primitives

3. Inference Engine, or Classifier consists of three separate modules. They are: group-determinator, classifier for confusing cases and classifier for peculiar cases of numerals Group-determinator based on the existence and position of primitives determines whether an input character is a confusing or a peculiar case. Confusing cases are identified as a member of one or two group(s) of subclasses. The inference engine for confusing cases checks all the rules of the selected group(s) of subclasses against all the extracted facts (features). An aggregation formula determines the total points scored by each subclass of the selected group(s) of subclasses. Classification is done by comparing scores against each other, Minimum-score threshold and Contention threshold values. The inference engine for peculiar cases determines the identity of an input character by examining its primitives and their locations

A block diagram of this recognition system is shown in Figure 1.2.

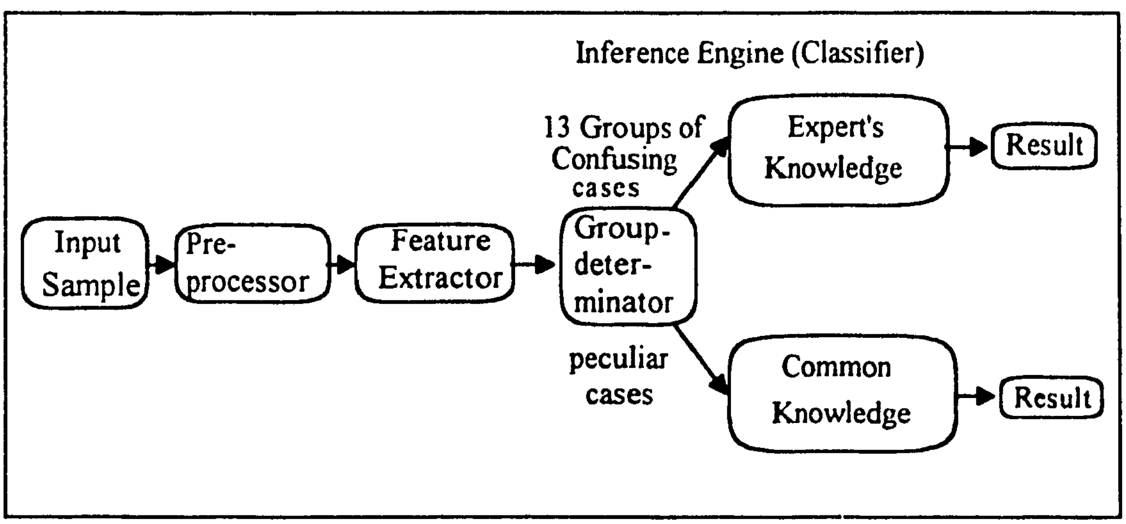


Figure 1.2: Block diagram of the proposed numeral recognition system

1.5 Outline of This Thesis

On the basis of character recognition skills of humans, a new recognition system is implemented and tested in this study. This recognition system is trained and tested on different sets of totally unconstrained handwritten numerals obtained from the ZIP codes of dead letters provided by the US post office.

Chapter 2 describes and analyzes the method used for acquiring human knowledge in recognizing handwritten numerals. It also analyzes the subjects' behaviours and performances.

Chapter 3 presents the knowledge elicitation procedure. This includes organization of knowledge and knowledge base, assigning weights to features and experts and conflict resolution method.

Chapter 4 briefly covers the preprocessing operations and in more details explains the feature extraction process.

Chapter 5 discusses classification (or inferencing) mechanisms used for peculiar (simple) and confusing cases of numerals. Overlapping of groups of subclasses, determination of scores of subclasses and threshold values are the major topics of this chapter.

Chapter 6 uncovers the source of test data and its validation procedure. It also explains the aggregation formula and presents the results of training and testing of the system on different sets of data. Then these results are analysed.

Chapter 7 makes a critical analysis of the methodology, design and implementation techniques used for building the different components of this numeral recognition multi-expert system. It also suggests enhancements and direction for future studies.

Chapter 2

Knowledge Acquisition

It is obvious that a knowledge based system can not perform well with inadequate, imprecise or erroneous knowledge. Therefore it is extremely crucial to have an appropriate method of acquiring knowledge. The method used depends on different factors, such as; availability of experts, domain of expertise, knowledge acquiring skills of the knowledge engineer, and so on ([32]).

The most common methods of fact-finding in system analysis are observation and sampling from existing records, interviews or group discussion and questionnaire or forms. These tools could be used in knowledge acquisition too ([15], [29] and [30]).

Capturing differing viewpoints to bear on a given problem has its advantages. Getting a group of experts to provide expertise on a given problem is a tricky issue. The problem is compounded when the experts disagree on some aspects of the problem. There are special tools for acquiring knowledge from multiple experts, for example MEDKAT ([31]) is one of them by which experts interact without direct contacts.

Consistently, there has been emphasis on group interaction, discussion and brain storming for acquiring knowledge from multiple experts ([31], [33], [34] and [31]) which is unfortunately lacking in this work. This deficiency is further discussed in the next sections of this chapter.

2.1 Method

The knowledge that is used in this expert system was gathered by the character recognition group of CENPARMI during their work on recognition of confusing cases of handwritten numerals ([13] and [27]).

They selected 360 (mostly difficult and some simple) cases of handwritten numerals from a database of 17,000 segmented digits from US zip codes. Examples of these confusing samples are shown in Figure 1.1. Table 2.1 shows how many samples from each class of numerals are present in the set of 355 samples There are only 355 and not 360 because five of the samples were found invalid because they had been placed upside down

Class	Number of samples
0	45
1	28
2	35
3	32
4	31
5	23
6	41
7	49
8	36
9	35
Total	355

Table 2.1: Number of samples from each class of numerals.

These digits in their original size in a form as shown in Figure 2.1 were presented to 9 human subjects. Five of them were experts (researchers) in character recognition and the others were students

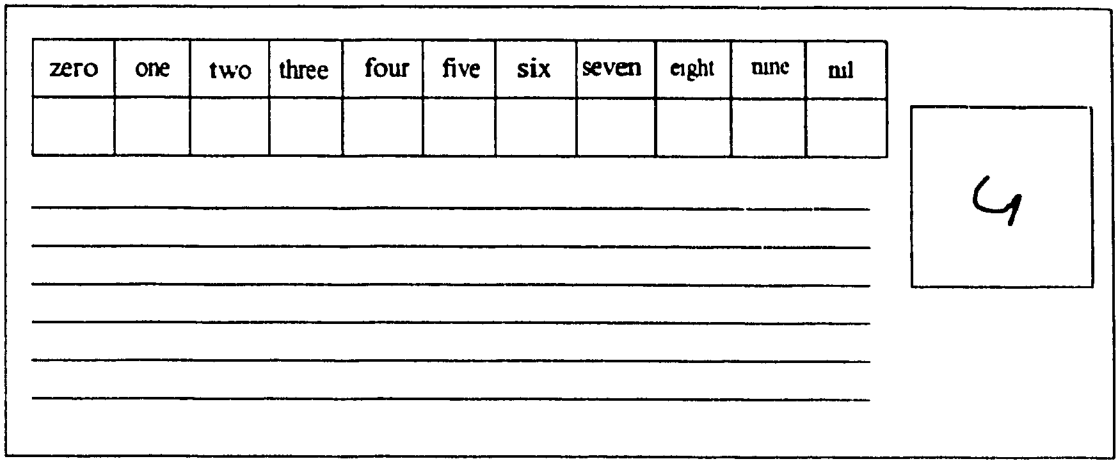


Figure 2.1: Knowledge acquisition form.

The forms for 360 samples and the following instructions on how to fill them were given to each subject.

1 360 samples of handwritten digits are to be identified
2. The experiment is conducted in 4 sittings, 90 samples for each sitting
3. Goals of this experiment are; first to find out which poorly written digits are confusing and which ones are not, second to learn on what basis humans actually make their decisions which may be used for machine recognition of numerals.
4. Pay attention to one sample at a time.
5. If you have no doubt about the identity of the sample, simply enter '100' in the box corresponding to your choice

6 If you hesitate between 2 (or more) possibilities, enter 2 (or more) numbers, totalling 100, in the boxes corresponding to the probable identities of the sample. A higher score indicates what is more probable to you.

7. The 11th box, marked NIL, should be used when you think the sample does not really look like a digit.

8 After identifying the digit, explain the key factors which lead to your decision(s). Try to write them down in decreasing order of importance.

9. A sample answer is provided to help you understand the procedure.

Subjects filled their knowledge acquisition forms without any discussion with one another or with those who conducted the experiment. Also there were no later clarifications about or corrections in the knowledge (comments) that had been gathered.

Expert subjects had had greater exposure to the problem of recognizing characters Hence they had a better understanding of what they should look for and put down on their forms. Experts had a better motivation than (little) money for going through this exercise. Generally, experts are expected to do a better job. Also there are figures supporting the argument that experts performed better than students ([13] and [27]).

In order to rely on quality knowledge for the knowledge base of this numeral recognition multi-expert system, we decided to use the knowledge that had been provided by the expert subjects only.

2.2 Interpretation and Organization of Experts' Comments

In order to elicit the knowledge that had been gathered from five experts, it had to be checked for redundancies, errors and other mistakes. To do so, all the comments of five experts had to be rewritten and reorganized in a more appropriate order.

Rewriting of the comments was necessary because on many occasions experts were saying the same thing but in different ways or words Hence a knowledge dictionary covering all the necessary words and definitions for commenting on the features of numerals was created. Therefore a common vocabulary was used to rewrite or translate all the comments (for some examples see Table 2.2). This was specially important because each of the experts had a different capability in expressing his/her thoughts in English Some of the definitions of this knowledge dictionary can be found in Table 4 1.

All knowledgeable comments (see Figure 2.2 for some knowledgeless ones) totalling 3,277 were entered in a database. This database was reorganized such that all comments of five experts with respect to a specific sample would appear under one another, making it easier to elicit knowledge for a specific case or class.

2.3 Contribution and Performance by Experts

Experts neither contributed nor performed similarly. Figure 2.3 and Table 2.3 show the number of comments written by each expert. For 355 samples expert number three wrote only 310 comments, leaving many of the samples with no comments Where as expert number one did his best in identifying more of the important features

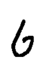
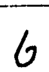
Character	Expert	Comments with the Original Wording	Same Comments with the Knowledge Dictionary Words
	1	Curve on top looks more like a 9	Curvature of top part of top cavity is toward north.
	1	Stem at bottom is relatively straight	Almost straight stem at the bottom
	4	Right concavity followed by left concavity.	East facing cavity on top. West facing cavity at the bottom.
	4	Width of top stroke indicates a "5".	Top side of top cavity facing east is long.
	1	Stem taller than the circle part	Top of the stem is the top of the character.
	1	Loop not closed.	A north facing cavity at the bottom.
	4	Oblique stroke	A stem with south-east Direction.
	4	Almost closed loop on the right side.	Opening of the cavity at right is small.
	1	Curly enough, loop shape.	A round cavity.
	4	Since the loop is on the left, the curly digit can be a "0".	A round cavity with a span ratio less than one (i.e., being a wide loop).
	1	Straight top.	A straight segment on top.
	1	Hook at bottom not long enough.	Length of hook at bottom is not long.

Table 2 2. Examples of experts' comments in their original wording and after translation using the words in the knowledge dictionary.

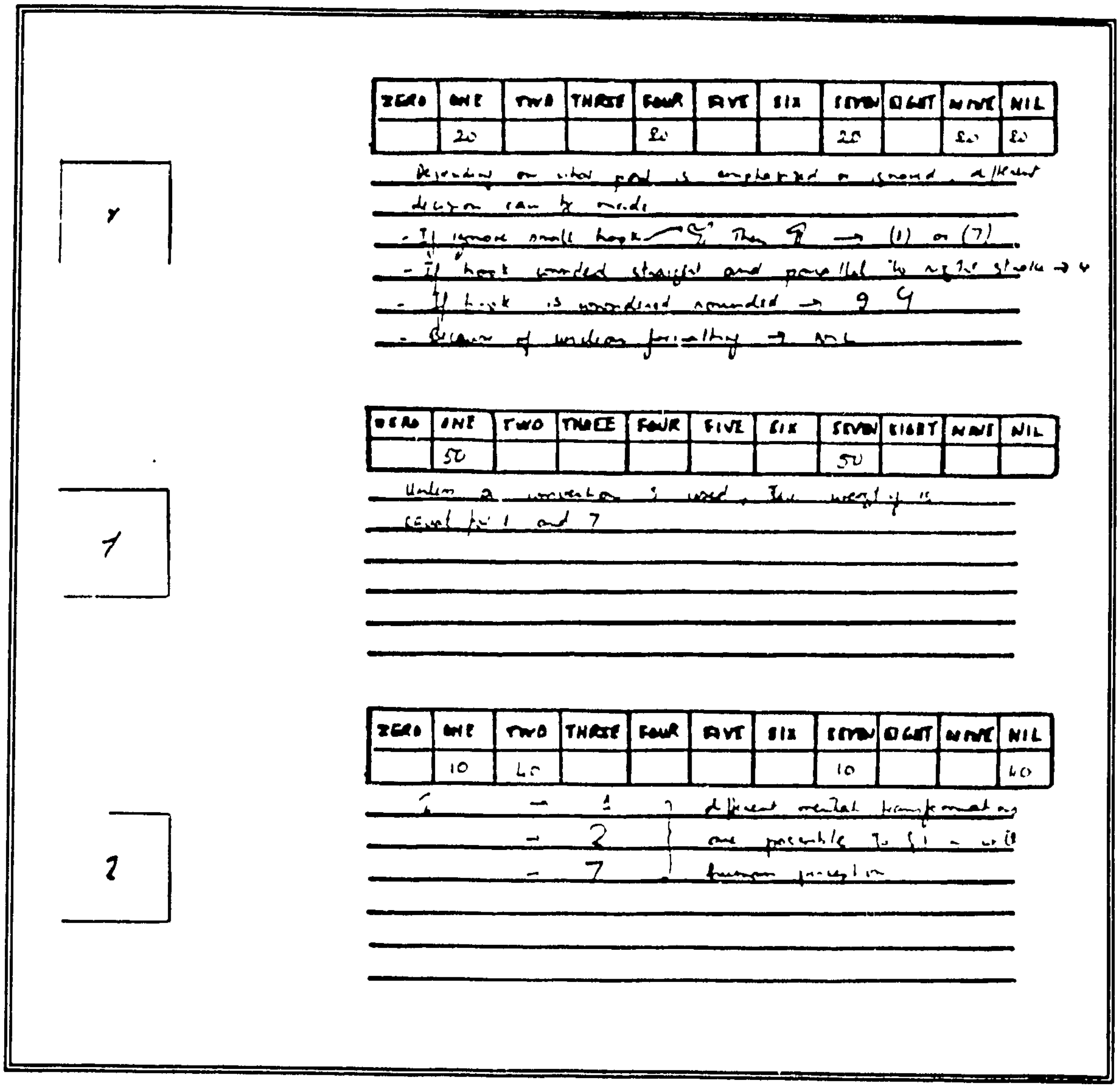


Figure 2.2. Examples of knowledgeless comments.

Comments of different experts are different in terms of quantity and quality. Figure 2.4 shows comments of different experts for the same sample, with respect to quality and quantity of the comments, expert no. 1 is the best, while expert no. 2 is the worst in this particular case.

Human Experts	Number of Comments
Expert No. 1	1,358
Expert No. 2	718
Expert No. 3	310
Expert No. 4	408
Expert No. 5	433
Total (All Experts)	3,227

Table 2.3 Number of comments provided by five experts.

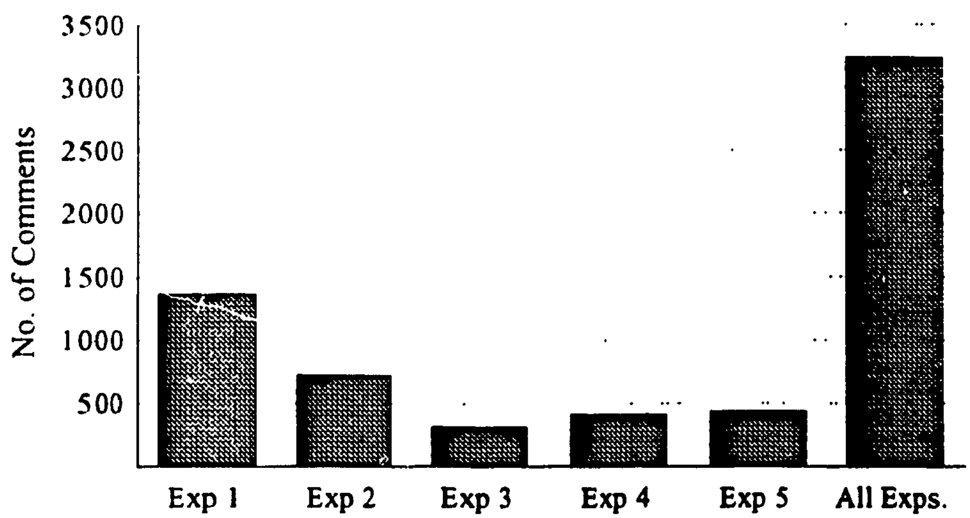
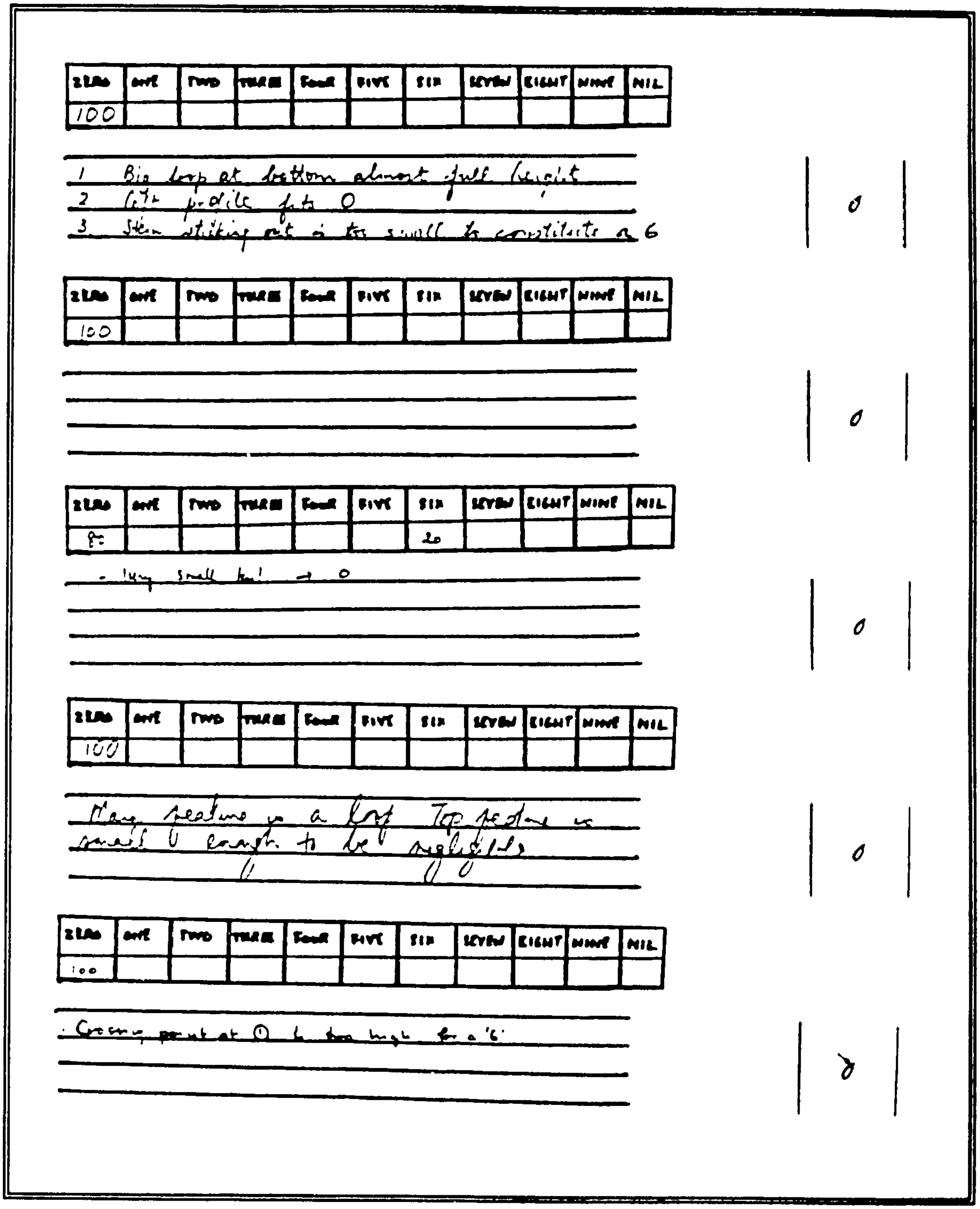


Figure 2.3: Graph of the number of comments provided by five experts.



ZERO	ONE	TWO	THREE	FOUR	FIVE	SIX	SEVEN	EIGHT	NINE	NIL
100										

1 Big loop at bottom almost full height
2 Little pedicle for 0
3 Stem sticking out is too small to constitute a 6

ZERO	ONE	TWO	THREE	FOUR	FIVE	SIX	SEVEN	EIGHT	NINE	NIL
100										

ZERO	ONE	TWO	THREE	FOUR	FIVE	SIX	SEVEN	EIGHT	NINE	NIL
80						20				

- Very small tail → 0

ZERO	ONE	TWO	THREE	FOUR	FIVE	SIX	SEVEN	EIGHT	NINE	NIL
100										

Many features for a loop. Top feature is small enough to be negligible

ZERO	ONE	TWO	THREE	FOUR	FIVE	SIX	SEVEN	EIGHT	NINE	NIL
100										

. Crossing point at ① is too high for a '6'

Figure 2.4: Comments by different experts on the same sample.

Figure 2 5 and Table 2 4 show the performance of different experts in recognizing 355 confusing cases of numerals. Real identity of the samples was determined by the majority vote, see Section 2.4. A certainty of 70% is enough to recognize (or substitute) the identity of a numeral, below that it is a rejection Expert number two, four and five have similar performances. Expert number three is too cautious and expert number one with a very low substitution rate has recognized up to 36% of the confusing numerals.

Human Experts	Recognition	Substitution	Rejection
Expert No. 1	36.1%	1.7%	62.3%
Expert No. 2	63.9%	4.2%	31.8%
Expert No. 3	16.9%	0%	83.1%
Expert No. 4	62.5%	6.2%	31.3%
Expert No. 5	58.3%	4.2%	37.5%
All Experts (average)	47.5%	3.3%	49.2%

Table 2.4: Human experts performances with 70% certainty.

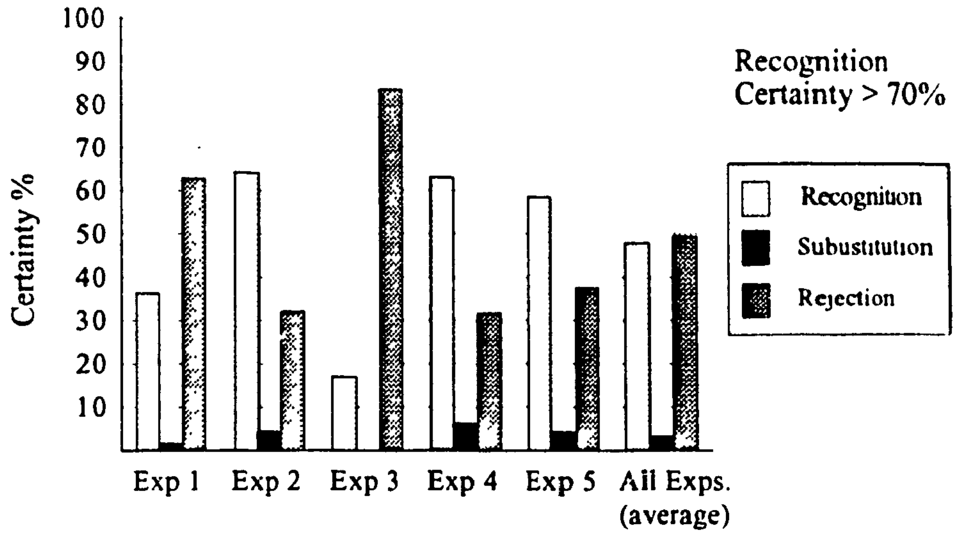


Figure 2.5: Graph of the performance of five human experts.

2.4 Real Identity of Samples

What are the real identities of the 355 samples that were used in the knowledge acquisition phase? Legault and others in their work ([13]) refer to the misclassification of numerals when the data base (including these 355 samples) was originally constituted.

We reestablished the identities of the 355 samples by majority vote method among five experts. Sixty samples (around 17%) were found to have different identities, Table 2.5 shows some of these samples with their original and new (real) identities.

Sample	Old I.D.	ID given by voting among all experts
	1	8
	6	0
	1	0
	5	3
	7	1
	4	6

Table 2.5: Some of the samples with their old IDs and new ones given by experts.

Chapter 3

Knowledge Elicitation

What is required from the elicitation stage is a complete and correct description of the expert's knowledge, and the way in which he/she handles that knowledge ([30]). Knowledge elicitation is a notoriously difficult activity. It becomes even more difficult when it is dealing with complex, unstructured or ill-formulated type of knowledge ([32]).

The power of expert systems derives from the quality of the knowledge in it. An appropriate presentation of knowledge is essential for having quality knowledge Impartiality of the knowledge engineer is another prerequisite for having correct and unbiased knowledge ([36]). Well conducted elicitation often reveals facts that were not previously at the forefront of consciousness of experts ([32]), see Section 3.1.

Eliciting knowledge from a group of experts is a major task since it has the additional difficulty of integrating different opinions ([14], [34]) There are many tools and techniques for knowledge elicitation which are mostly domain dependent ([33] and [36]).

3.1 Sorting out Rules for Each Class of Numerals

A total of 3,227 comments, by five experts, regarding the features of only 10 characters (numerals 0 to 9) had to be checked for duplications, contradictions and other errors and finally put into a format usable by the inference engine (or classifier) of a multi-expert numeral recognition system.

We started the elicitation process by putting the relevant comments (features) pertaining to each class of numerals under its own category. However, very soon it was realized that some of the comments (features) that were noted for a variation of a class would not hold for the other variations of the same class. Different variations of the same class of numeral might even have contradictory features Therefore further classification of classes of numerals into their subclasses evolved which was a matter of necessity rather than choice.

3.2 Subclasses and Groups of Subclasses

What is the basis of sub-classification of classes of numerals? Comments and identity percentage assignments by experts reveal three types of information about each sample. They are;

1. Feature(s) that distinguish(es) one sample from another in the same or different classes of numerals.
2. Each variation (subclass) of a class resembles the variations (subclasses) of other classes.
3. Importance level of each feature.

The first type of information distinguishes samples from one another and hence results in the creation of subclasses. The second type of information identifies similar subclasses and leads to the formation of groups of subclasses. Finally the third type of information determines the weight of each feature.

We continued the elicitation process and thereby we found forty eight subclasses of numerals in thirteen groups. These subclasses and their groups are shown in Figures 3.1 to 3.8.

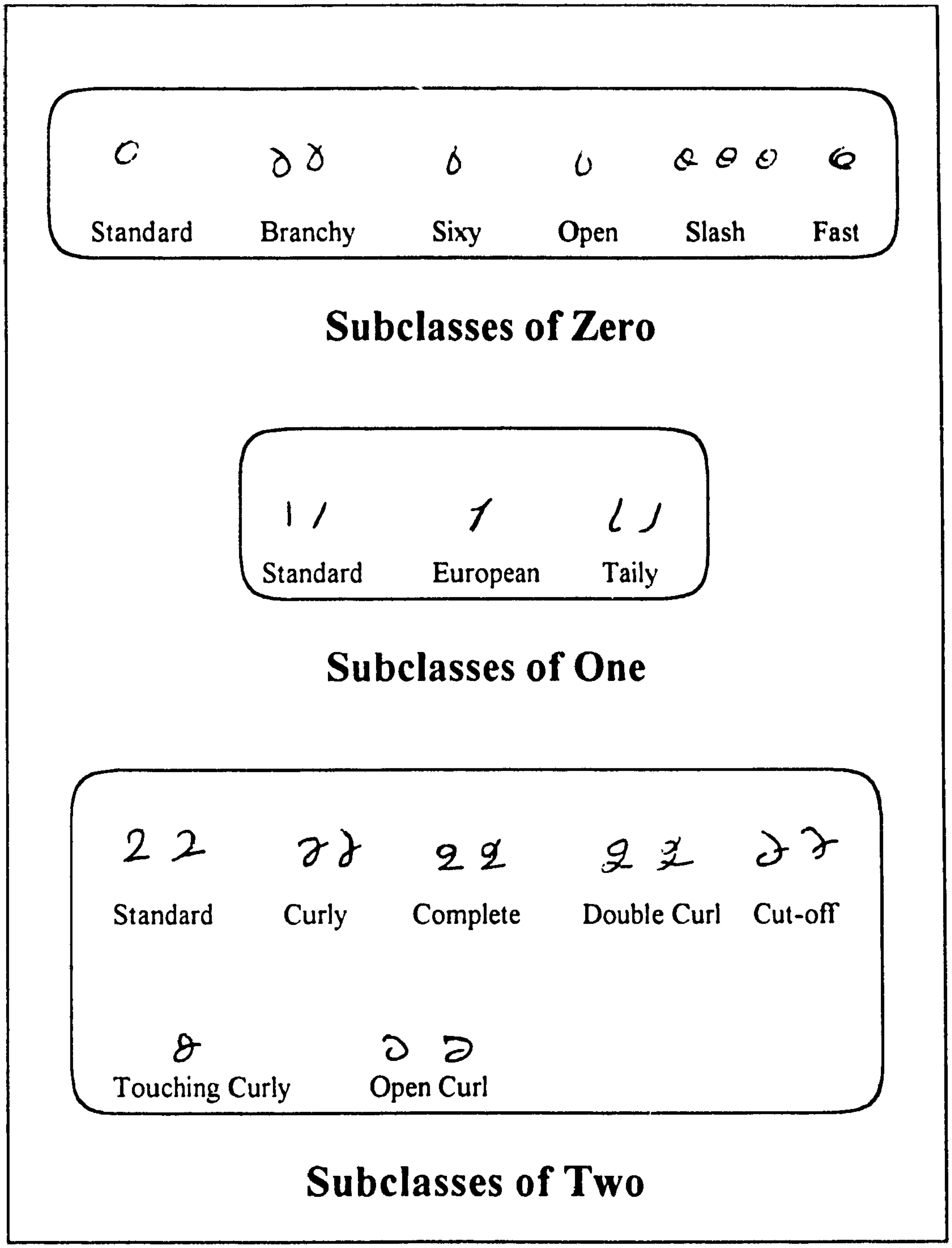


Figure 3.1: Subclasses of classes 0, 1 and 2

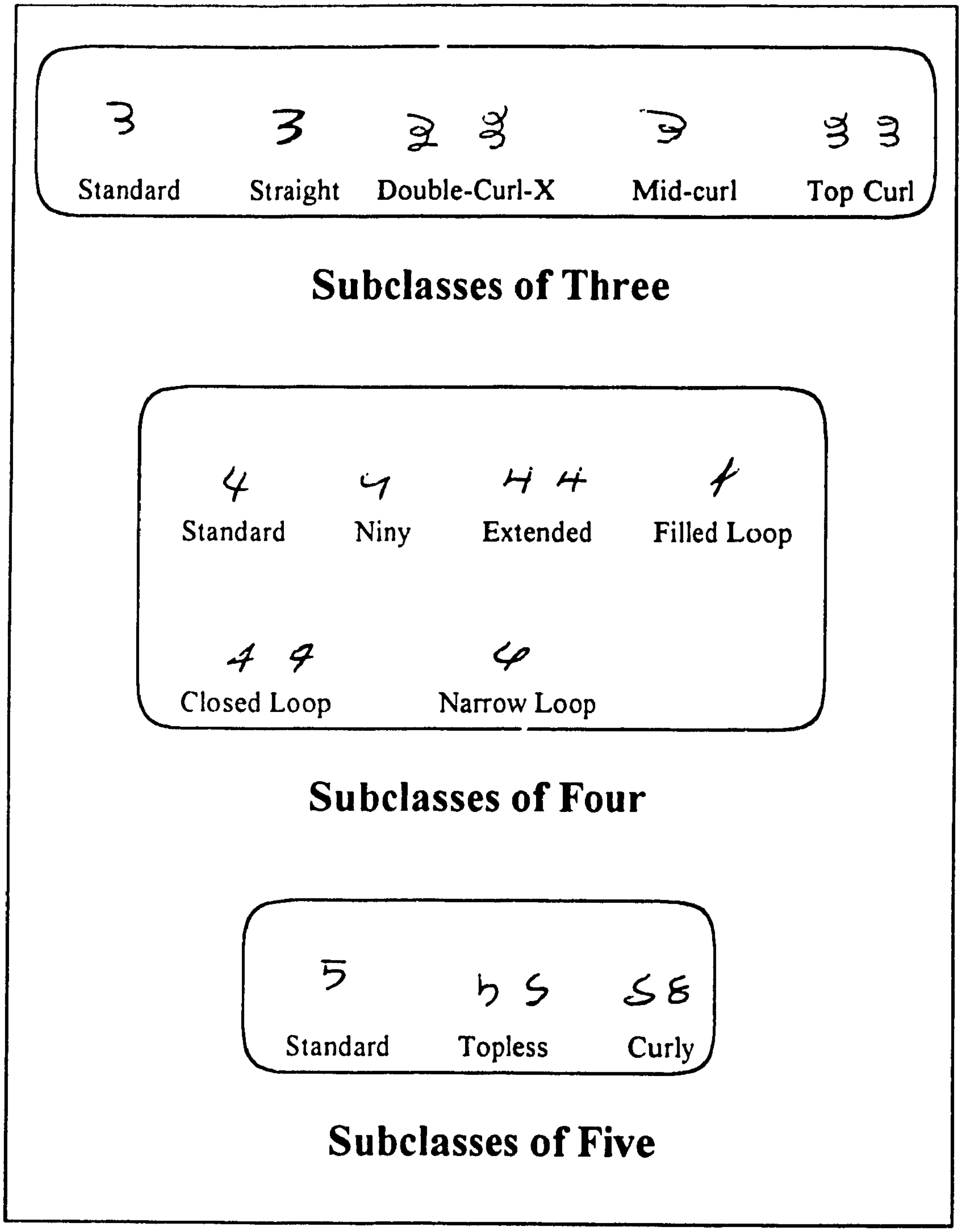


Figure 3.2: Subclasses of classes 3, 4 and 5

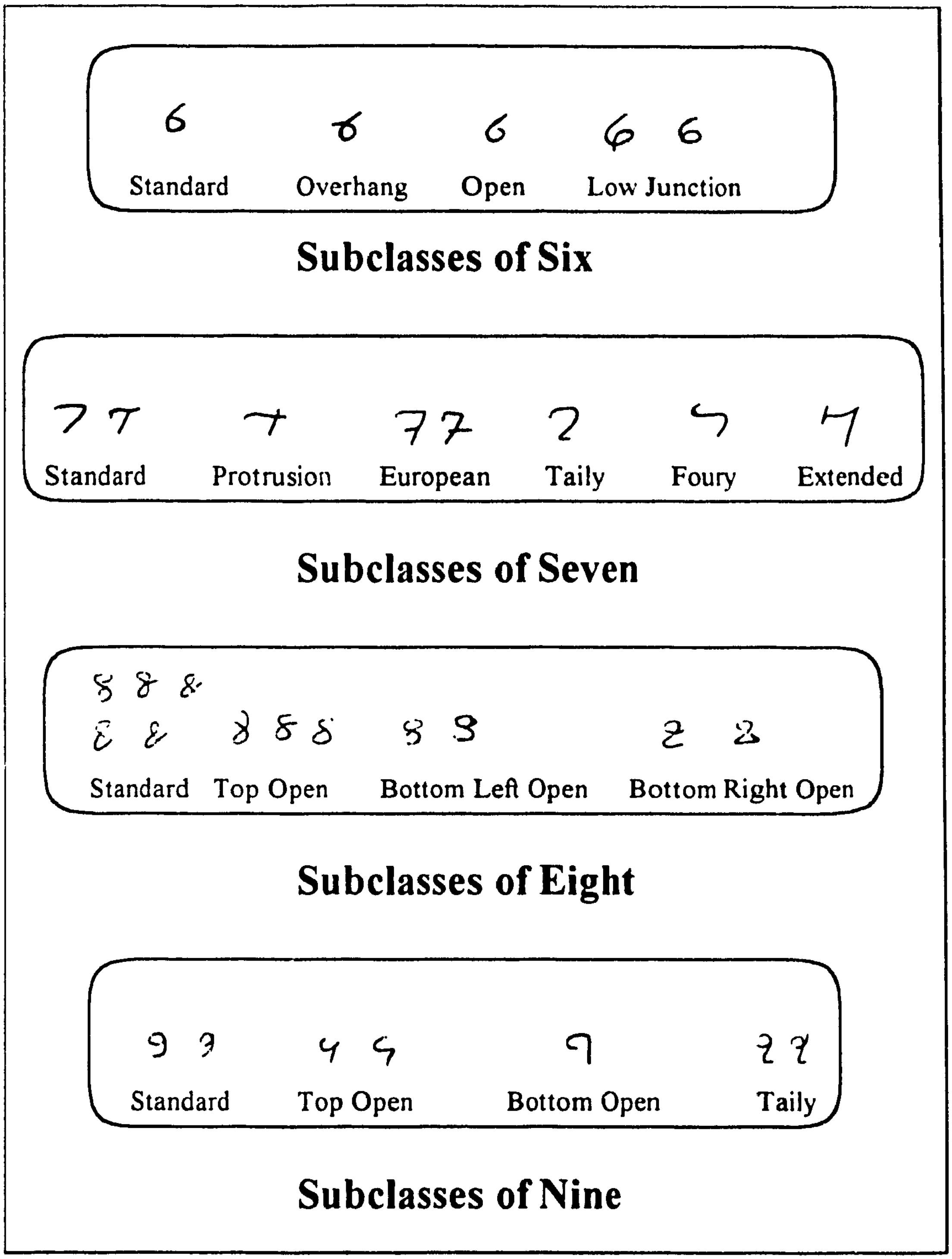


Figure 3.3 : Subclasses of classes 6, 7, 8 and 9

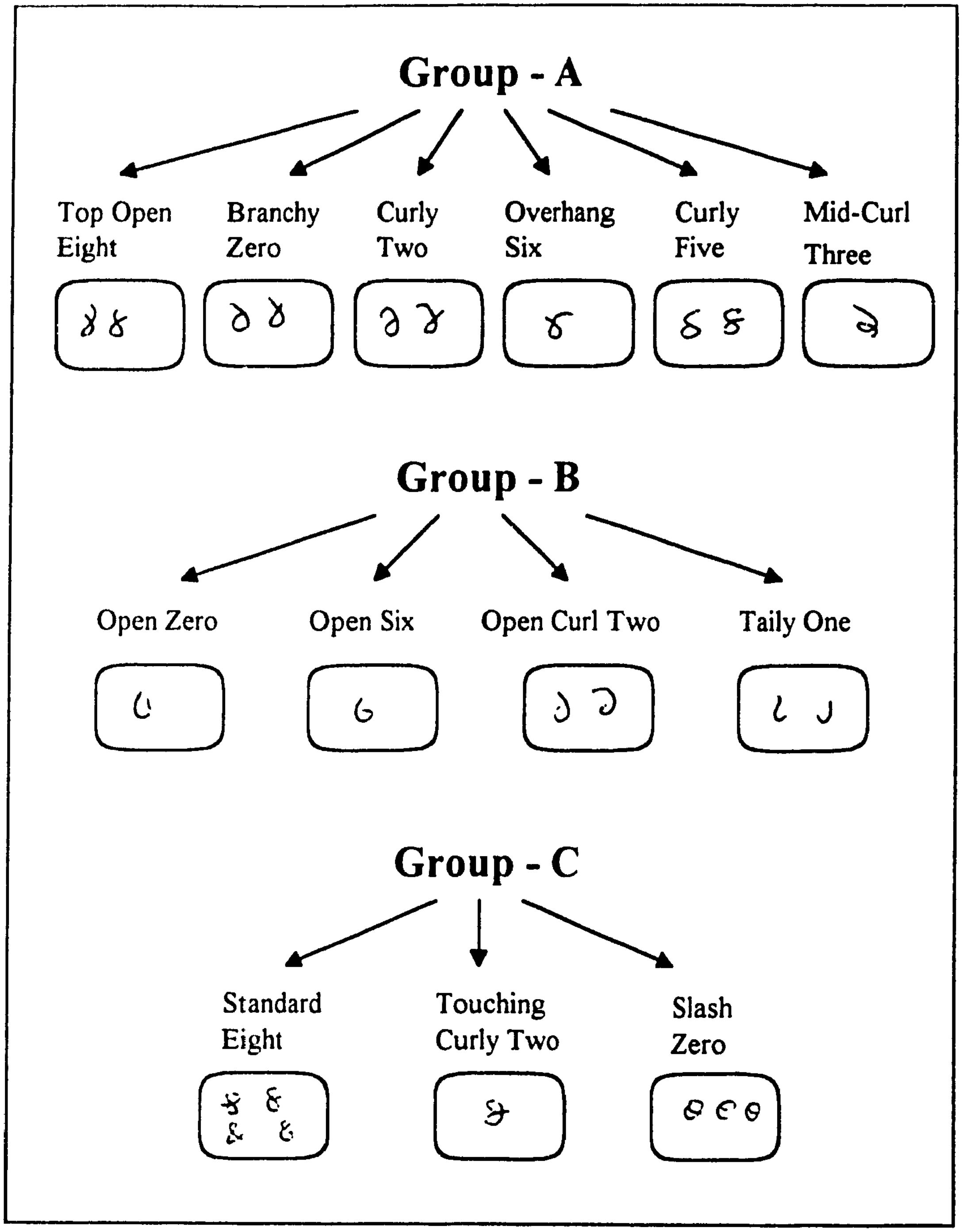


Figure 3.4: Members of subclass groups A, B and C

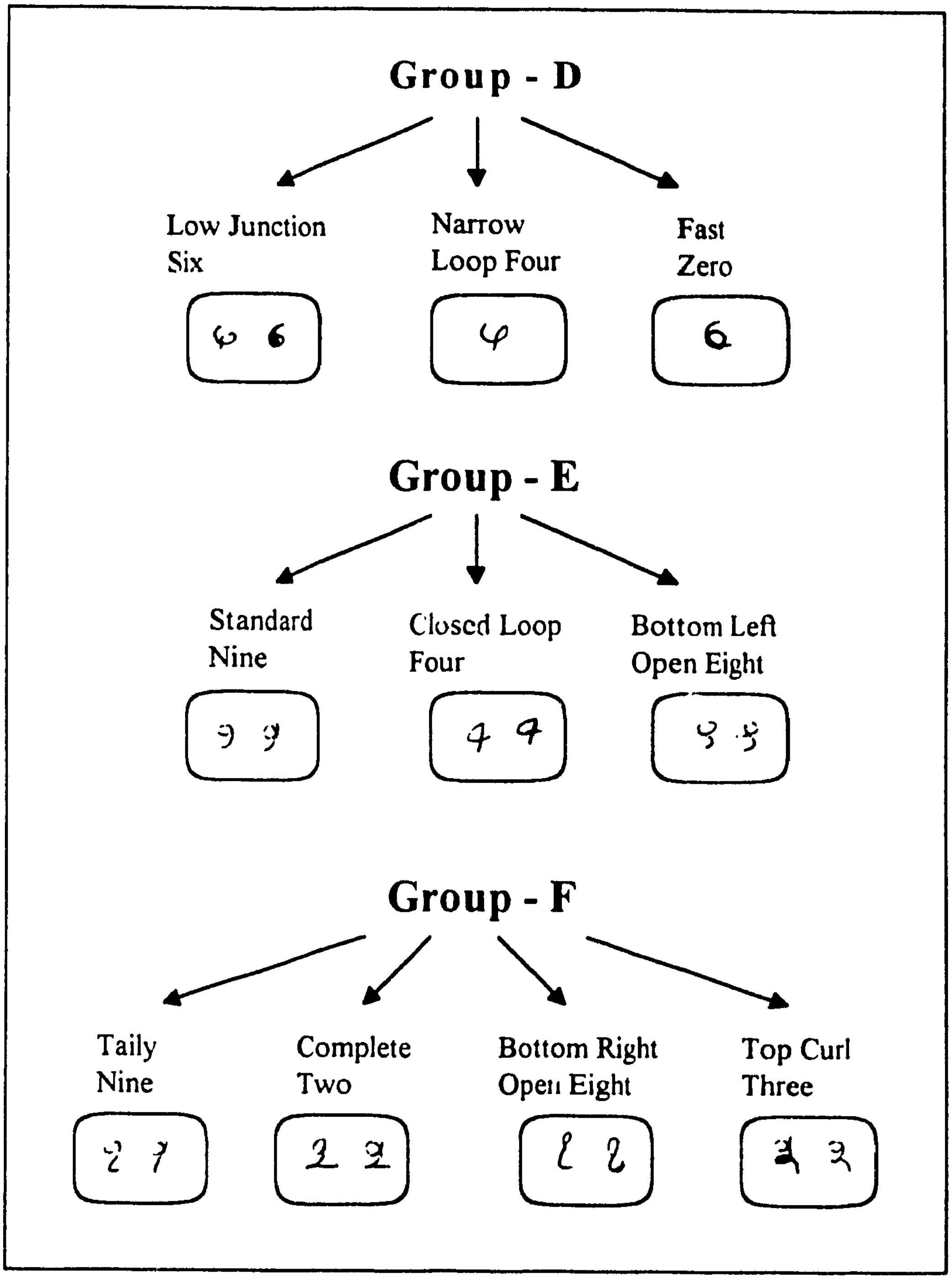


Figure 3.5: Members of subcclass groups D, E and F

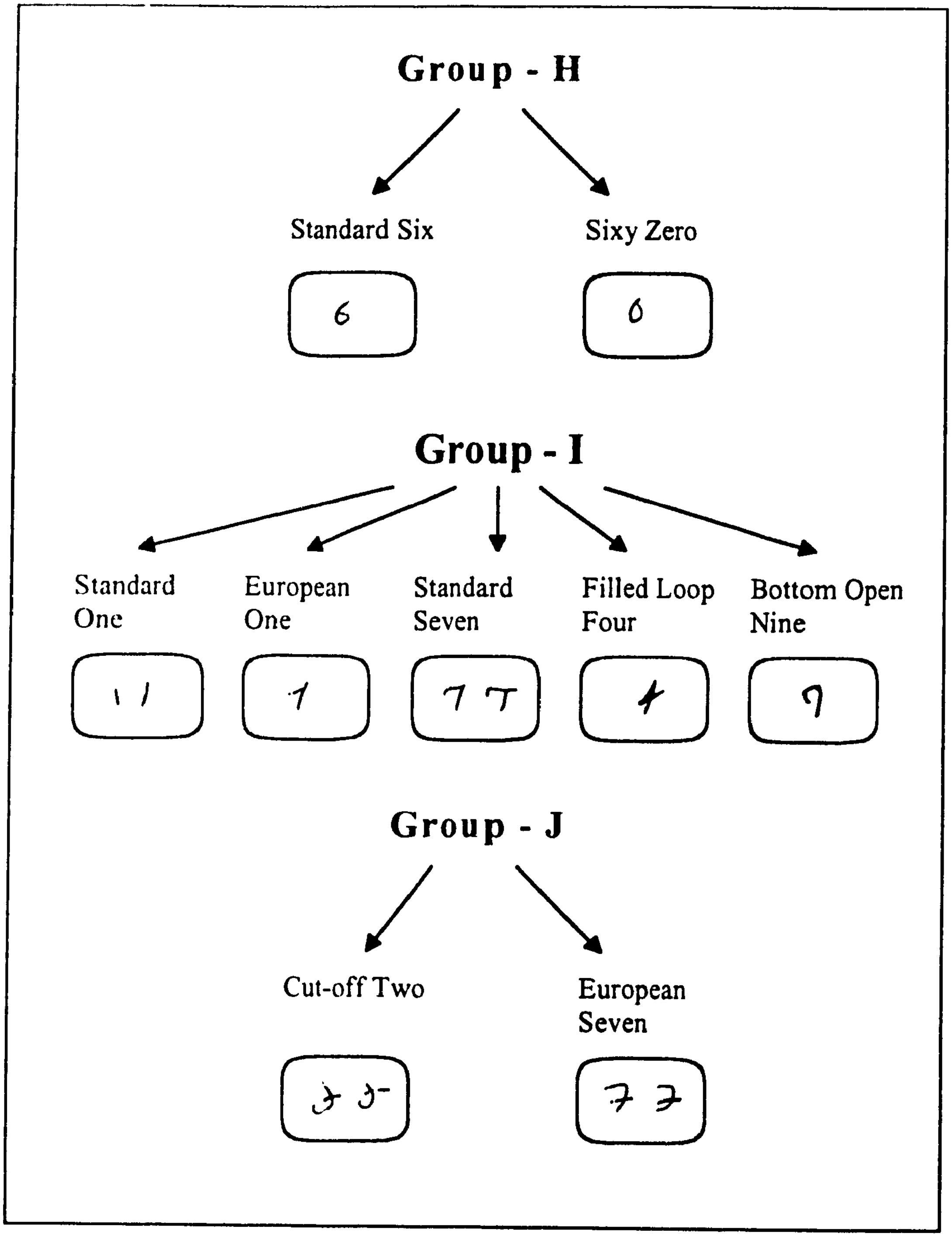


Figure 3.6: Members of subclass gropus H, I and J

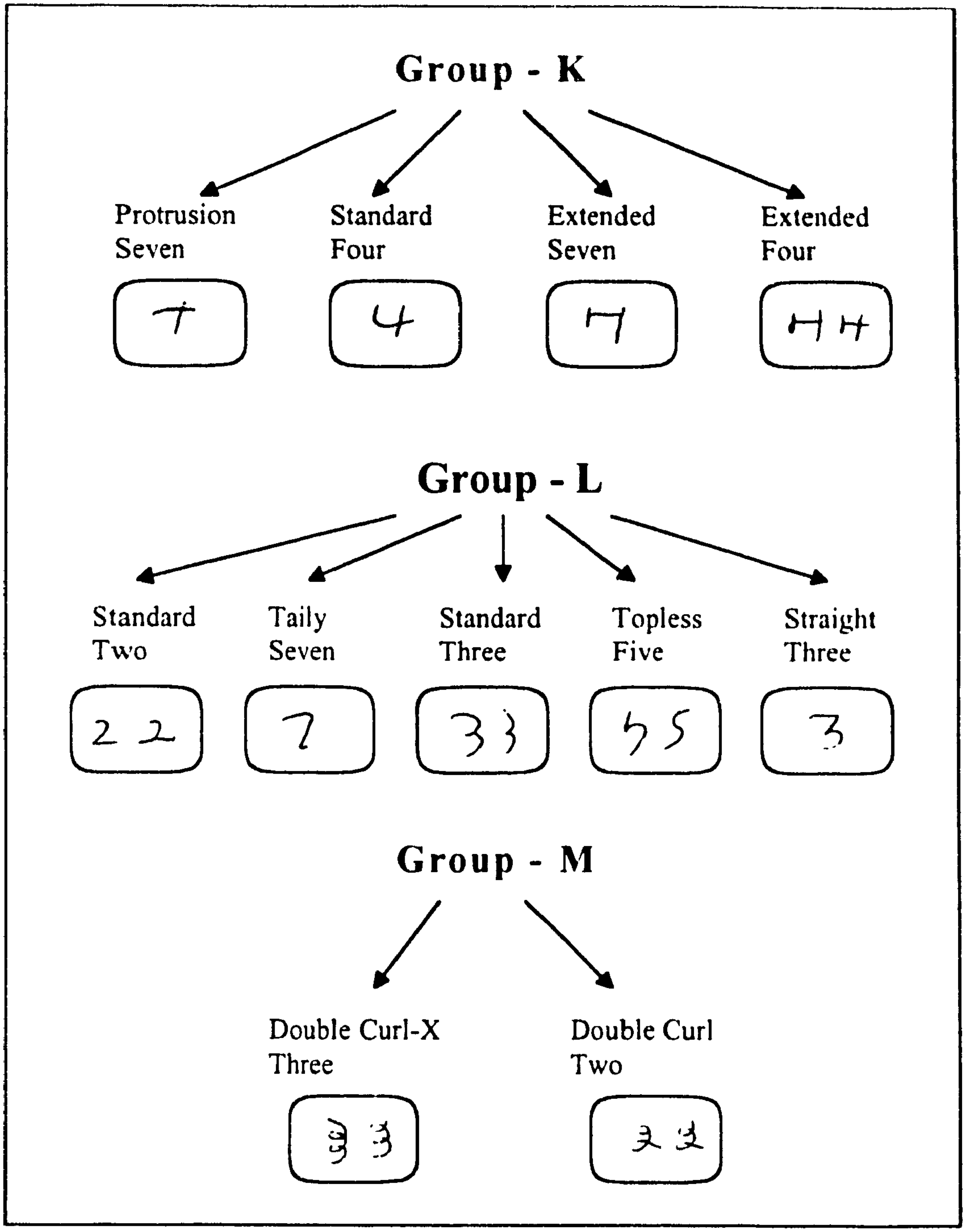


Figure 3.7: Members of subclass groups K, L and M

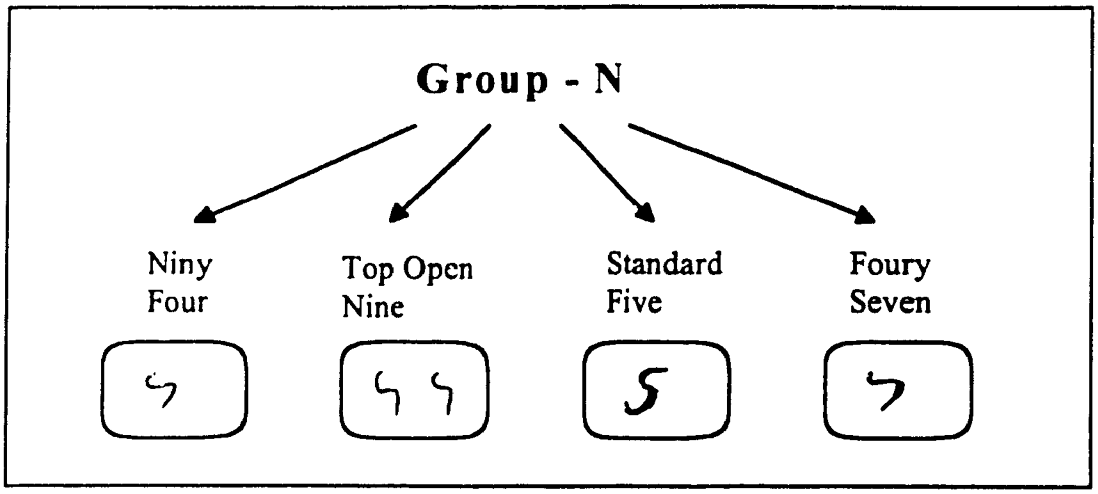


Figure 3.8: Members of subclass group N.

3.3 Impartiality, Fuzziness, Conflicts and Weight Issues

Impartiality

A knowledge engineer can introduce bias into expert judgement. Specifically, the data gatherers and analysts can cause bias through misinterpretation or misrepresentation of the expert knowledge ([36]).

Throughout the elicitation phase we maintained utmost impartiality in order to ensure the validity of the expert system. This was done by minimizing the alteration of comments and avoiding the use of any feature that was not provided by the experts.

Fuzziness

Experts wrote their comments about the features of the 355 samples without any restriction on the wordings of their comments Therefore the expert's comments inherited the inexactness (or fuzziness) of the natural language. For example, adjectives like short or long are used to describe the length of a line segment ([25] and [32]).

The (generally) imprecise comments of the experts were used (almost exactly as they were) in the construction of the knowledge base. Feature extractor (see Section 4.3) adds fuzziness to the extracted features and thereby making it possible to compare them with (fuzzy) rules in the knowledge base.

Conflicts

Experts do not possess the same knowledge, and this differing knowledge is used differently, even though all the experts received the same briefings and information as background to the problem ([36]). This is the cause of conflicting opinions among experts.

Experts participation at the knowledge elicitation phase (or generally interaction among experts) can resolve such conflicts to a great degree ([31, [33] and [34]). Lacking the presence of experts and the need to remain impartial required us to incorporate even the conflicting opinions of different experts in the knowledge base There is a negative effect on the total score of any subclasses that embodies conflicting comments (see Section 5.3.2).

Weights

As instructed (see Section 2.1), experts were required to write down their comments in the order of importance. Hence a line code to indicate the line number of a comment was attached to each comment. Also an expert code for indicating the author of a comment was attached to each comment. Comments that were provided by more than one expert got the expert code "0" and the highest line code for such comments was used as their line codes. Table 3.1 shows comments, expert codes and line codes for the subclass "standard three". Depending on the author and line number the weight of a comment varies. Section 5.3.2 discusses the weighing mechanism in more detail.

3.4 Creating a Knowledge Base

Sub-classification of classes of numerals resulted in the dispersion of the comments of a class under its different subclasses. Some of these comments were definitely valid only for one subclass while others could be equally relevant to the other members (subclasses) of the same class. Hence comments under subclasses of each class were checked if they were valid for the other members of the same class and if so they were duplicated under other member(s) of that class

Features (rules) for all subclasses after going through different stages of correction, refinement, dispersion and compilation were put together to form a knowledge base This knowledge base embodies all the features (rules) of all subclasses which are necessary to verify if a sample is a subclass known to this knowledge base

The created knowledge base consists of more than 350 rules (features) covering 13 groups of subclasses or 48 individual subclasses. Figure 3.9 shows some of the segments of the subclass 'standard three' and Table 3.1 shows all its rules (features).

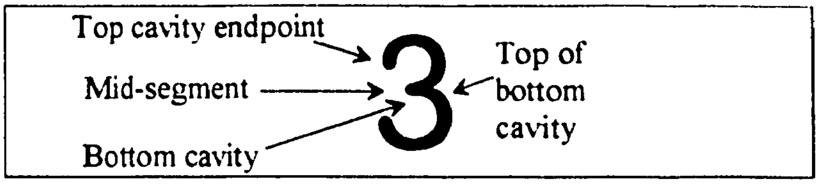


Figure 3.9: Definition of some of the segments of the subclass standard three.

No.	Rules (Features)	Expert Code	Line Code
1	Mid-segment clearly exists.	1	1
2	Indentation on the right of mid-segment clearly exists	1	1
3	There exists a West facing cavity at the bottom that is round in shape.	0	1
4	There exists a West facing cavity on top that is round	0	1
5	Direction of lower endpoint of bottom cavity is west.	2	1
6	Direction of lower endpoint of bottom cavity is South-west	1	2
7	Mid-segment is horizontal	0	1
8	Directions of all three endpoints (top, bottom, mid-segment) are West.	5	1
9	Top portion of bottom cavity is not very short	0	1
10	Top portion of bottom cavity is a line with a slope about +1.	5	2
11	Top endpoint and bottom endpoint are almost on a vertical line.	2	2
12	Top portion of top cavity is not very short.	5	2
13	Direction of strokes changes as that in a three.	5	1
14	Existence of two clear west facing cavities.	4	1

Table 3.1: All of the rules (features) for the subclass 'standard three'

Chapter 4

Preprocessing and Feature Extraction

4.1 Noise Removal, Skeletonization and Normalization

Depending on the type of features that must be extracted from a pattern, different preprocessing operations are needed. Binarization, normalization, rotation, region growing, skeletonization, contour detection and edge smoothing are some of the preprocessing operations ([21] and [23]).

This recognition system uses binarized patterns of numerals as input. Binarization and skeletonization of the input patterns had been done by ([19]), see Figure 4 1 However some of the patterns still contained noise in the form of isolated points which were removed by the preprocessor module of the system

In a recognition system that is built on the basis of human recognition of characters, every bit of the pattern counts. This means that even a very short stroke can be important for correct classification of a character ([44]). Therefore, the very existence of constructs of a pattern, their sizes or orientations should not be altered during the preprocessing phase. This ensures the integrity of the pattern and reduces the preprocessing time.

This system is very sensitive to noise while distortion, style variation , size differences and small degree of rotation are less disturbing. Noise, such as an extra segment can cause rejection or substitution of a character for obvious reasons. Variation in the size of the input characters is not important (i e., there is no need for normalization) because all the measurements are made relative to the length of the diagonal of a rectangle enclosing the skeleton of the input character. For example, if the length of a segment is less than 30% (smallness coefficient) of the length of the diagonal then its relative length becomes "small". These coefficients are pre-determined and their values are determined during the training phase (see section 6.4) in order to have the most suitable values.

A bit of rotation also does not change the extracted features. Because direction of segments are determined in a crude way. For example, direction of a cavity can be north-east instead of a specific angle (see Table 4.1).

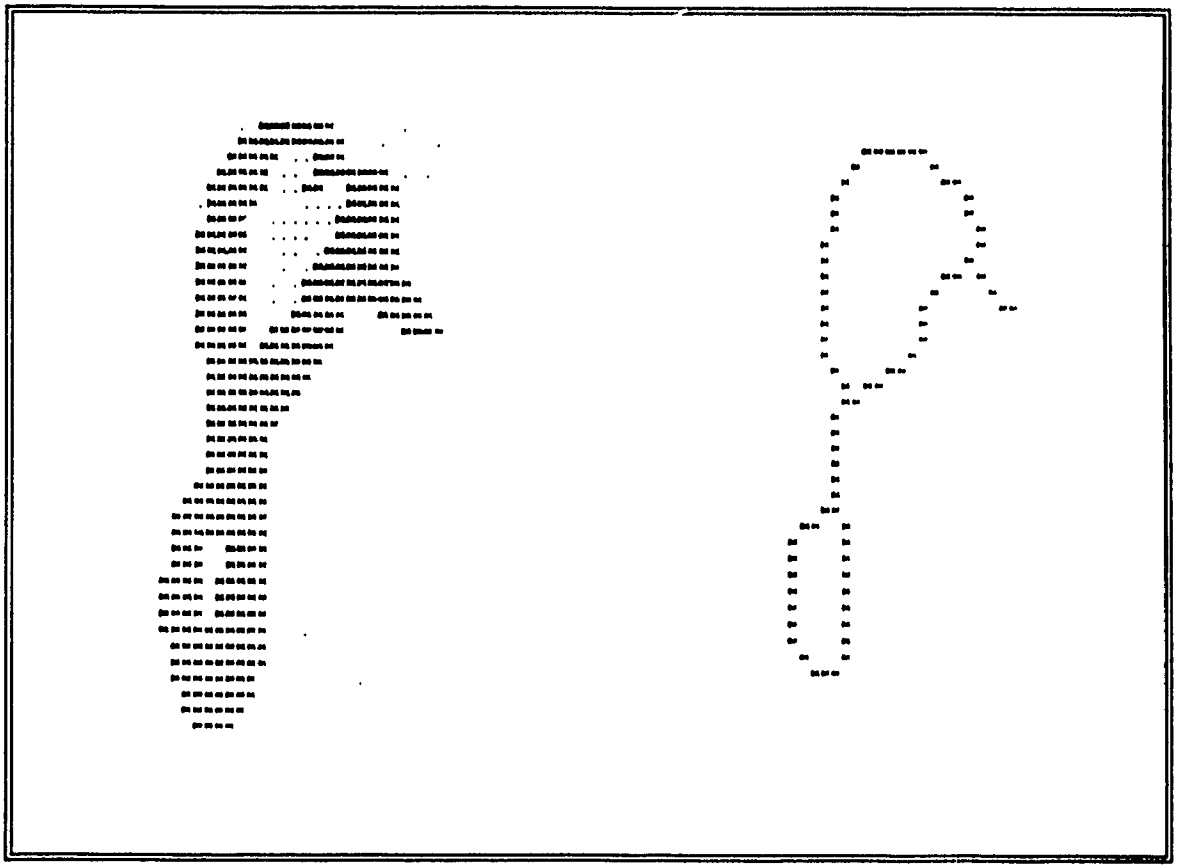


Figure 4.1: Examples of binarized numerals with their skeletons.

4.2 Feature Selection

Blesser et al ([40]), have argued that the failures of previous feature detection techniques lies in the ad hoc nature in which the features were chosen. It was concluded that the features must have some psychological significance so that the recognition algorithm can exhibit the required flexibility ([18]).

Generally, geometrical and topological features are used by human for identification of characters ([16], [22] and [23]). Hence this recognition system too must extract this kind of features from a numeral. Among many geometrical features extractable from a character the system needs to detect only those which have been identified by the five human experts in their comments, see Section 2 1 From experts' comments we compiled a list of all the necessary features which were to be extracted, most of them are shown in Table 4 1

4.3 Feature Extraction

Some researchers believe that the primary problem of recognizing handwritten characters is feature detection and not feature specification ([22]). Depending on how features are extracted a classification method might work better or worse ([16]).

In this recognition system, different feature extraction methods can be used to extract different types of features. Though we have mostly used the skeleton of a character for extracting its features.

Feature extractor scans the skeleton of an input sample from top to bottom and determines its endpoints and junction points. This breaks down the skeleton of a sample into segments Each segment is terminated by endpoints

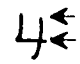
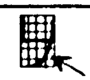
No.	Segment Features	Line	Curve	Cavity	Loop	Figure
1	Absolute length: The actual length of the skeleton of a segment	X	X	X	X	 A segment
2	Relative length e.g., Short, Long, ... which is determined in comparison with the length of the diagonal of the matrix of the character	X	X	X	X	Diagonal 
3	Number of points. Number of points which form a segment	X	X	X	X	
4	Distance between endpoints: Shortest distance between beginning and end of a segment	X	X	X	X	
5	Type of segment Line, Curve, Cavity or Loop by comparing the ratio of the absolute length and the distance between endpoints of a segment to a predetermined value called "type threshold".	X	X	X	X	 A cavity  A curve
6	Number of rows and columns: Number of rows and columns of the matrix of a character which are spanned by a segment	X	X	X	X	 3 rows and 3 cols
7	Span ratio: Number of rows of a segment divided by its number of columns (see point 6 above)	X	X	X	X	
8	Extreme points: Row and column of the extreme top, bottom, left and right points of a segment.	X	X	X	X	 Extreme right point

Table 4.1: Features extracted from the segments of a character

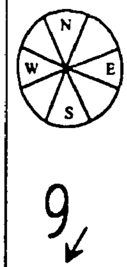
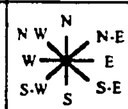
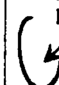
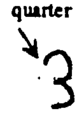
No	Segment Features	Line	Curve	Cavity	Loop	Figure
9	Direction Direction of the segment is the slope of the lines or curves connecting their beginning point to their end point. For cavities it is the slope of the line connecting the turning point of the cavity to the middle point of the line between the end points of the cavity Then this slope value is marked as North, North-East, ... on the basis of predetermined ranges of slope values	X	X	X		 Decent has S-W direction
10	Start point and end point directions Determined by the direction formed by the last two points at the end points of a segment	X	X	X	X	
11	Slope Slope of the direction of the segments (see point 9 above)	X	X	X	X	
12	Depth Length of the line from the turning point of a cavity or curve to the middle of the line connecting their end points		X	X	X	Depth line 
13	Vertical and horizontal positions· Position of a segment relative to the height and width of the matrix of the character. Resulting in Left, Middle, . positions.	X	X	X	X	 On the bottom and right
14	Start and end points vert & horz. positions. As in point 13 above, but for the end and the start points of a segment.	X	X	X	X	
15	Start and end points quarters Positions of the start and end points of a segment as in a cartesian co-ordinate system. Centre of the system is the centre of the matrix of the skeleton of the character.	X	X	X	X	2nd quarter 

Table 4 1· Features extracted from the segments of a character. (Continued)

No.	Segment Features	Line	Curve	Cavity	Loop	Figure
16	Composite or stand alone segment A segment is composite when it is formed by more than one segment, e g., a loop can be composite Single segments are stand alone segments			X	X	 3 composite loops can be formed
17	Indices of the neighbouring segments Segments that share at least one point are neighbours.	X	X	X	X	 Neighbours
18	Number of junctions: Segments can have a maximum of two junctions	X	X	X	X	
19	Direction of curvature. Curves are bent Up, Down, Left, .. , which is the direction opposite to the direction of their depth line See point 12 above		X			 Bent South-West
20	Opening relative length and direction Relative length (Short, Medium, ...) and direction (North, North-East, ..) of the line joining the endpoints of a cavity Also see points 2 and 9 above			X		
21	Horz and Vert. opening relative length Relative length (Small, Medium, .) of a special opening of a cavity, see the picture in this row			X		
22	Angle, shape and turning point Relative position of the turning point of a cavity (Up, Middle, Down, ...) and with the turning point as the vertex determining the angle between the sides of the cavity. Shape of cavity is determined on the basis of the angle of cavity (Round, Sharp, ...)			X		 Sharp  Round

Table 4.1: Features extracted from the segments of a character (Continued)

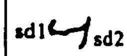
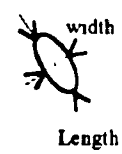
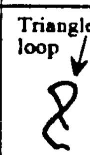
No	Segment Features	Line	Curve	Cavity	Loop	Figure
23	Loop tendency A round cavity (point 22 above) with a relatively small opening (point 20 above) resembles a loop			X		
24	First side and second side detail features: Relative length, relative position, number of points, degree of curvature, ... of the sides of a cavity are determined as it is done in a complete segment			X		
25	Loop length and width. Loop length is the length of the longest line that can be fit inside a loop, and loop width is the length of the line perpendicular to the loop length line and crossing at its centre				X	
26	Straight bottom and top That is when a predetermined percentage of the points of a loop or cavity form a line at the bottom or top portion of a segment			X	X	Straight line on top 
27	Sharp bottom and top When the number of points that form the top or bottom 10% of a loop is less than a predetermined number, the loop is sharp at top or bottom respectively				X	
28	Triangle and round shapes A loop with a straight bottom and a sharp top or a straight top and a sharp bottom forms a loop triangular in shape				X	
29	Bottom slope and quarter Slope and (cartesian) quarter of the bottom portion of a loop Bottom portion of loop is determined relative to the height of the loop similar to point 26 above.				X	

Table 4.1 Features extracted from the segments of a character. (Continued)

or junction points. Next these segments are processed to determine their type, i e, if they are lines, curves, cavities or loops ([19] and [28]).

Each kind of segment has its own specific types of features which must be determined, see Table 4.1. For example, only in case of cavities, the features of the sides are determined as well. Figure 4.2 shows a binarized sample with its skeleton before and after tracing. In segmented skeletons, endpoints are indicated by '*', junctions by '#' and cavity turning points by '+'. Finally the general features of the sample (like its span ratio) are determined

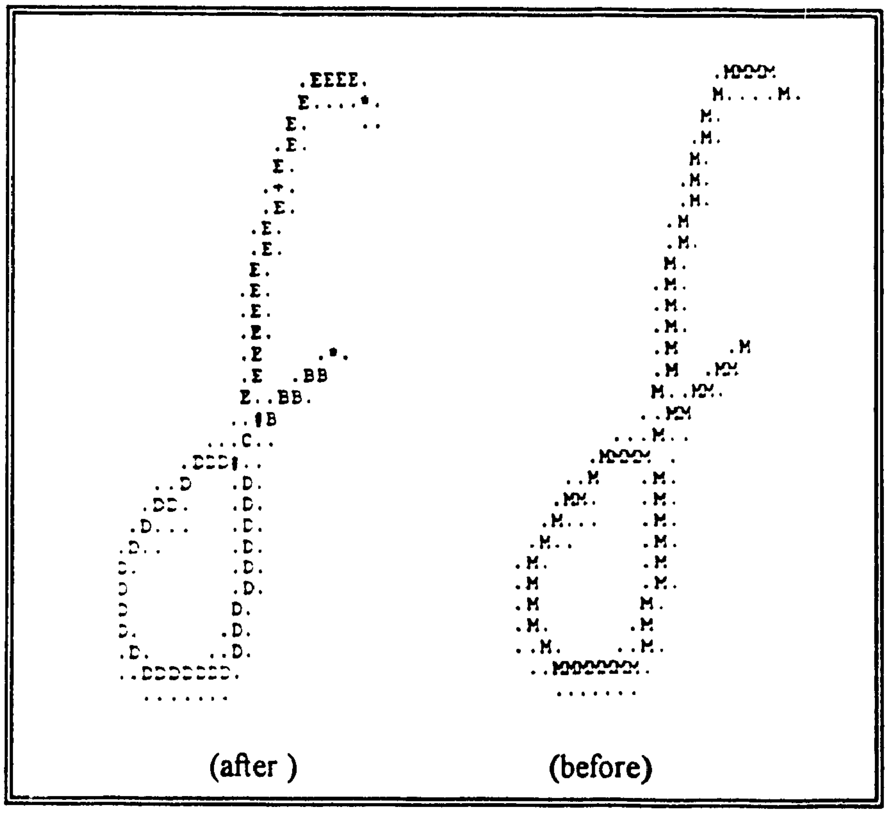


Figure 4.2: A sample before and after tracing.

Using the skeleton of a character to extract its features has the advantage of simplification, as it reduces the pattern to different types of lines only

However this advantage is (usually) accompanied with the problem that the skeleton does not fully represent the sample ([38]). For example.

1 Loops have become lines (Figure 4.3 - a).

2 Angles between segments are distorted (Figure 4.3 - b).

3 Junction points are distorted (Figure 4.3 - b).

4 New non-existing segments are created (Figure 4.3 - c)

One way of alleviating these problems is to use different methods to extract features For example, loops, junction points, and angles between segments can be determined more precisely from the contours of the samples ([26]), see Figure 4.4. However the use of multiple methods of feature extraction has a negative effect on the processing time

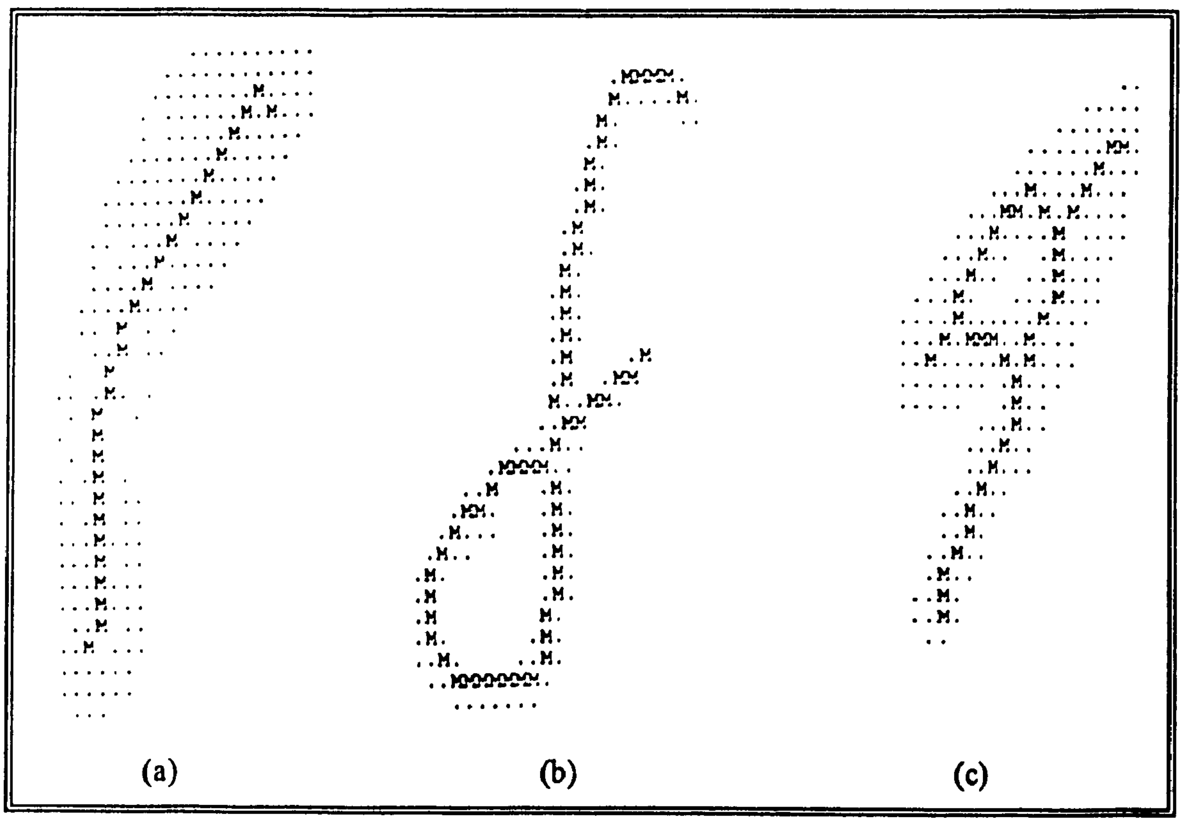


Figure 4.3: Distortions due to skeletonization.

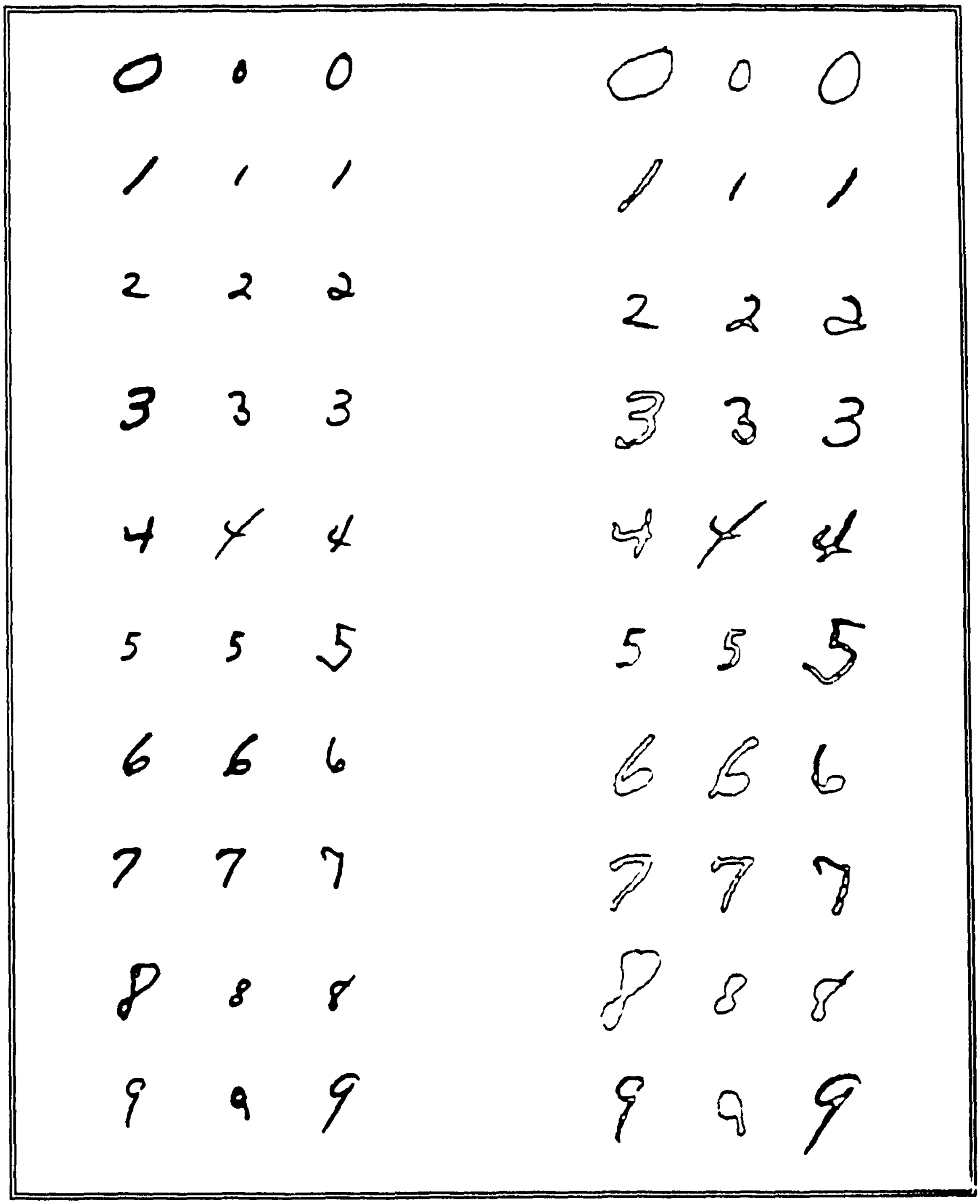


Figure 4 4: Using contours to determine angles and junctions

Chapter 5

Inference Engine (Classifier)

The two basic components of an expert system are; the knowledge base that must contain lots of high-powered knowledge about the problem domain and an inference engine containing knowledge about how to make effective use of the domain knowledge ([7])

According to K Parsaye, inference is the process of combining facts and rules ([6]) There is no simple, general way to characterize an inference engine How it should be structured depends on both the nature of the problem domain and the way in which knowledge is represented and organized in the expert system

In this multi-expert recognition system, the Group-determinator module and the classifiers for peculiar and confusing cases of numerals together form a backward chaining inference engine ([1] and [7]). The inference engine begins its search after the feature extractor module completes the extraction of all the facts (features) from an input character.

5.1 Group-determinator

As shown in the block diagram of this recognition system (Figure 1.2 - reproduced on page 43) a module called "group-determinator" determines if a sample belongs to any of the 13 confusing groups or not. This decision is made on the basis of the features extracted from the sample Belonging to a confusing group or not determines which of the two possible classification processes will be carried out next.

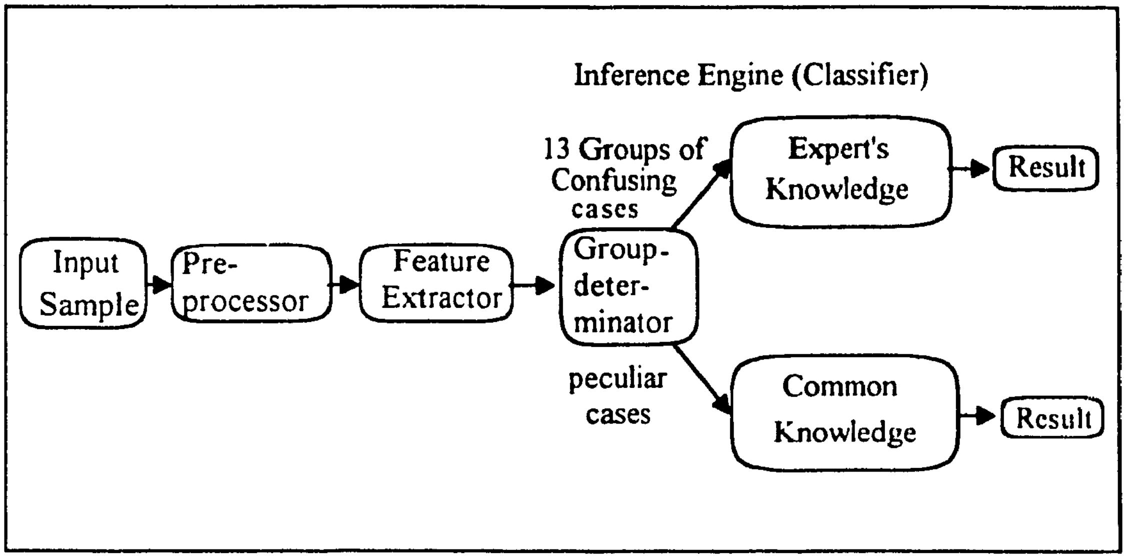


Figure 1.2: Block diagram of the proposed numeral recognition system

Samples which do not fit into (or belong to) any group are usually those which are written in a peculiar way. This peculiarity makes them distinct and naturally easy targets for classification. Their unusual shapes (constructs) can be detected easily and then identified. Figure 5.1 shows some of those samples that we call "peculiar cases".

Numeral	Peculiar Shape
Eight	
Five	
Four	
Nine	

Figure 5.1: Numerals with peculiar shapes.

Well written numerals and not that well (carefully) written numerals fall into the category of confusing cases of numerals, and they may belong to one or two groups of subclasses. This category comprises most of the numerals that we usually encounter, see Section 5.4.

5.2 Overlapping Groups

How could a confusing numeral belong to two groups of subclasses (see Section 5.1)? Groups of subclasses can be considered as sets with features of their respective subclasses as their members. Subclasses of different groups of subclasses exhibit similar features. Therefore groups of subclasses (as sets) may intersect and hence it is possible that a subclass is labelled as a member of two different groups of subclasses. For example a sample that looks like a '6' may be labelled as a member of Group - A and Group - H This enables the system to conduct a more detailed examination of the sample when necessary. See Figures 3 4, 3.6 and 5.2.

When a sample is labelled as a member of two different groups of subclasses, the system may come out with two different identities for that sample. Section 5.4.2 explains how this kind of conflict is resolved.

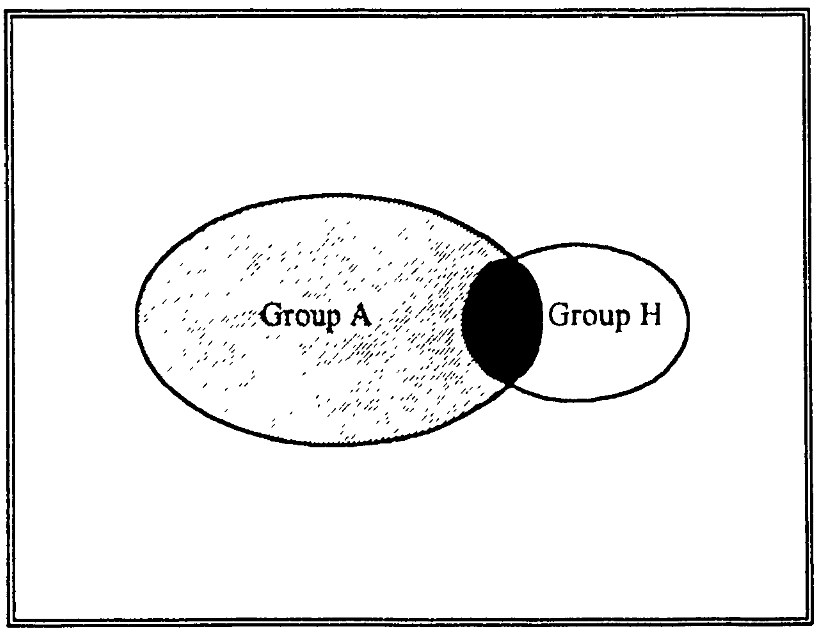


Figure 5.2: Intersection of two groups of subclasses.

5.3 Classification of Peculiar (Simple) Cases

Peculiar cases are very distinct in shape and hence do not cause any confusion with other cases Therefore all that is needed for a recognition system to identify a peculiar case is to verify the existence, position and relationship of the segments which form that character. This is a relatively simple operation that does not take much time. Therefore, the system saves time by processing peculiar and confusing cases differently.

In this recognition system, when a peculiar case is identified, it is believed to be true with 100% certainty Twenty five peculiar cases are defined in this recognition system Any peculiar case which is unknown to this recognition system is a rejection case

5.4 Classification of Confusing (Difficult) Cases

Group-determinator (see Section 5.1) checks the existence, position and relationship of the segments that constitute a character. Those samples that meet all the required criteria for being a member of one group are labelled as members of that group. The criteria for being a member of a group are set by us These criteria are very liberal and obstruct only those samples which have peculiar shapes or are filled with noise. In all our tests more than 70% of the samples qualified for membership in one or two groups of subclasses (see Section 5.2).

5.4.1 Firing of Rules and Score Aggregation

Depending on the group(s) label(s) of a sample, certain set(s) of rules from the knowledge base are selected to be tested if they can be fired These rules (features) are tallied against the facts (features) that were extracted by the feature extractor module, in case of a match the matching rule is fired.

For example, if a sample is labelled as a member of group H, all the rules of subclasses "standard six" and "sixy zero" are separately examined whether can be fired See Figure 3.6.

Firing of rules scores points A subclass scores more or less points depending on how many of its rules can be fired. The formula that determines the total score for any subclass is shown below.

$$s_s = \sum_{j \in u_s}^{u_s} e_j l_j m_s$$

Where·

s_s - Score of subclass s $\quad 0 \le s_s \le 100$

u_s - Set of fired rules of subclass s

e - Expert weight $\quad 0 \le e \le 1$

l - Line weight $\quad 0 \le l \le 1$

m_s - Maximum score for each rule of subclass s

m_s = 100 / no. of rules

When all experts are considered to be equally qualified, the Expert weight (e) for all experts must have the same value. Expert weights can be different for different experts, for example could be dependent on their previous performances. Sections 6.2 and 6.3 discuss this issue in more detail. Similarly, experts' comments over different lines of the knowledge acquisition form

(Figure 2 1) can have the same or different weights. In this system a higher line number means a lower line weight (l). More rules for a subclass means less maximum score (m_s) for each rule of that subclass.

When a rule is fired, the additional score obtained due to that is the product of its Expert weight, Line weight and Maximum-score $(e \times l \times m_s)$.

This aggregation method (formula) differs from some of the more usual multi-expert opinion combination mechanism ([36] and [37]). It differs as it maintains experts hierarchies, does not override minority vote and does not force any kind of consensus (since there has not been any).

Disagreement among experts or their indifference with respect to a feature of a subclass (which are both permissible in this method) have negative effects on the total score of that subclass. Low score for a subclass may result in a substitution or a rejection, see Section 5.3.3.

The following example is presented in order to show how the above formula is used Suppose for an input character, which is categorized as a member of subclass group - L, three rules as shown in Table 5.1 are fired.

Rule numbers of subclass "Standard three" that are fired.	Expert Code	Line Code
1	1	1
5	2	1
10	5	2

Table 5 1: Rules of subclass "standard three" which are fired (example).

Subclass "Standard three" has a total of 14 rules, therefore $m = 100/14 = 7.14$ Let weights (e) for experts 1, 2 and 5 be 1, .75 and .5 respectively. Also let weights (l) for lines 1 and 2 be 1 and 5 respectively.

For the first rule the score is = 1.0 x 1.0 x 7.14 = 7.1

For the fifth rule the score is = 0.75 x 1.0 x 7.14 = 5 4

For the tenth rule the score is = 0.5 x 0.5 x 7.14 = 1.8

Therefore its total score (s) = 7.1 + 5.4 + 1.8 = 14.3

5.4.2 Two Thresholds for Recognition (or Substitution)

Character recognition systems usually use a certainty threshold as the criteria for recognition or rejection of an input sample. It is obvious that a false recognition is a substitution ([2], [28])

In this numeral recognition system, two thresholds are used as the criterion for recognition or rejection of an input character. They are Minimum-score threshold and Contention threshold. The Minimum-score threshold can have values between 0 and 100 and the Contention threshold can have values between 1 and infinity. The following paragraphs explain how these thresholds are used.

When an input sample is labelled as a member of a group of subclasses, all the rules of all the subclasses of that group are examined if they can be fired This firing attempt is done for each subclass of that group separately. A total score (according to the formula in Section 5.3.2) is obtained for each subclass of a group that is under evaluation.

If the total score of a subclass satisfies all of the following conditions then the class identity of that subclass (right or wrong) is considered as the identity of the input character, if not, the system refuses to identify the input character

1. Total score of the subclass is higher than the Minimum-score threshold
2. Total score of the subclass is the highest score in its group of subclasses.
3. The ratio of the total score of the subclass (i.e., highest score) and the total score of the next lower highest score is greater than the Contention threshold.
4. When two groups are examined, the total score of the subclass must be the highest among the total scores of all the subclasses in both groups

Training and testing of the system were done with Minimum-score threshold of 30 and contention threshold of 1.2, see Section 6.3.

Chapter 6

Training and Testing of the Multi-expert System

6.1 Source of Data and its Validation

Three sets of samples of numerals (sets 'A', 'B' and 'T') were selected for training and testing of this numeral recognition multi-expert system Each of these sets had 200 samples of each class of numerals, i.e , 2000 samples in each set All these samples belong to a database of 17,000 samples that had been collected from dead letter envelopes made available by the U.S Postal Services. This ensures that test samples are written by different authors from different locations in the U.S A Figure 6.2 shows some of these samples.

As it was discussed in detail in Section 2.4, the identity labels of some of the samples in the set of 355 confusing cases were reestablished These corrections were extended to the data sets 'A', 'B' and 'T' for those samples which were members of the set of 355 confusing cases too. This was done in order to ensure the validity of the results or in general the validity of the multi-expert system ([6] and [39]).

6.2 Implementation Issues

This multi-expert numeral recognition system was implemented by compiling approximately 20,000 lines of code using the Turbo C/C++ compiler. On a 486/33MHz IBM compatible PC, the average time for classifying one character is 175 milliseconds.

Following is a description of Figure 6.1 which illustrates the working (steps) and implementation of this multi-expert numerals recognition system.

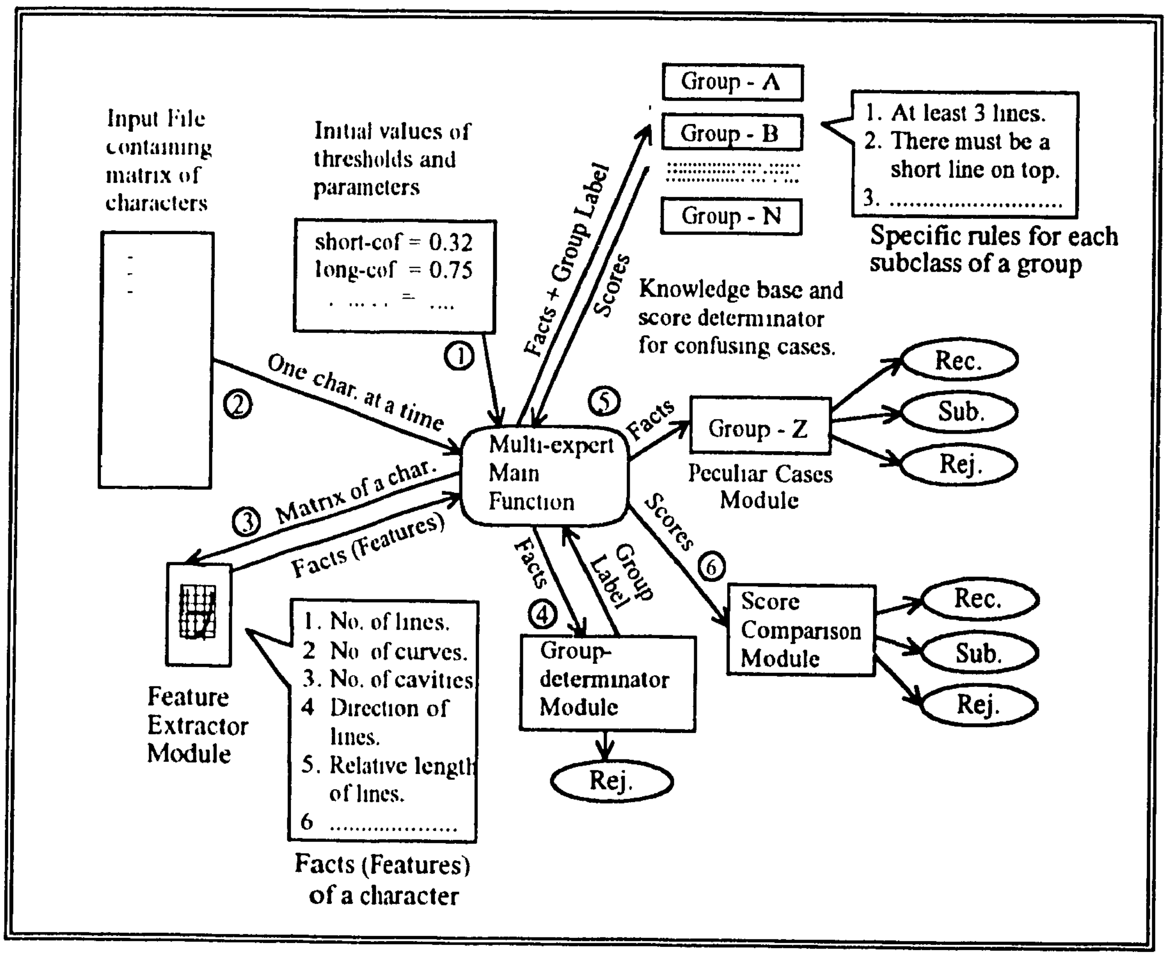


Figure 6 1 . Implementation and working of the multi-expert system.

Details of the design of each specific module (operation) can be found in its respective chapter.

Main function of the Multi-expert system (shown in the center of Figure 6.1) selects and invokes all the necessary steps or procedures that are required to determine the identity of a character. Encircled numbers in Figure 6.1 are associated with major modules that constitute this multi-expert system and their order corresponds to the order of activation of these modules

Binarized representation of characters (which must be identified) are stored in a file (example in Figure 4.1. The main function of the multi-expert system is to read these binarized characters into a matrix one at a time Then the feature extractor module traces the skeleton of the input character (using its matrix) and segments it into its primitives (lines, curves, cavities and loops)

A non-loop segment (primitive) becomes a line, a curve or a cavity depending on its degree of curvature. Degree of curvature of a segment is calculated by dividing its length by the distance between its endpoints. For example, a segment becomes a line if its degree of curvature is smaller than L_tsh (line threshold) which is a predetermined threshold set during training A typical value for the L_tsh is 1.1.

Other features of segments are extracted by analysing and synthesizing them further. For example, relative horizontal position (see Table 4 1) of a segment is "right" if 60% or more of its points are on the right half of the matrix of the character. All features found are recorded in order to be used by the inference engine of the multi-expert system.

Next, the main function invokes the group-determinator module (number 4 in Figure 6 1, also see Section 5 1) The group-determinator reads the recorded facts (features) and based on the existence and position of segments decides if an input character belongs to one or two known group(s) of subclasses (confusing cases) or belongs to peculiar cases or is unknown and must be rejected For example, if a character has only one cavity with an opening toward north, then it belongs to subclass group "B". In case of rejection, the main function reads the next character from the input file and starts the recognition process for that character

Depending on the group label of characters, their recorded facts (features) are further analysed either by the peculiar cases module or by the confusing cases module (number 5 in Figure 6.1)

The module which handles the Peculiar cases knows exactly what it is looking for, i e. specific number or types of segments and specific features in the existing segments (Figure 5.1) After it finds all that it is looking for, it comes out with a decision about the identity of an input character. For example, if it finds out that, in the recorded facts, there is only one segment and that is a loop, it decides that the identity of the input character is zero (a recognition or substitution possibility). When it does not find all that it is looking for, it refuses to make any decision (rejection).

When the group lable(s) of a character is one of the group labels that contains confusing cases (Section 5.4), the main function of the multi-expert system invokes the corresponding confusing case(s) module(s). Each confusing case module contains all the rules relevant to the recognition of that confusing case and the necessary inferencing mechanisms for checking those rules against the recorded facts An example of a rule for the numeral seven is that; is there a south-west facing cavity which has straight sides? For more examples of rules see Table 3 1

All the rules that are coded in an invoked "confusing case module" are checked (tallied) against all the recorded facts (by using IF and CASE statements). A subclass scores points when any of its rules finds a matching fact (or is fired) Depending on how many rules are checked and according to the aggregation formula in Section 5.4.1, each subclass of the selected group(s) of subclasses scores more or less points. There is an example in Section 5.4.1 which describes how the score of a subclass is calculated.

Finally the main function of the multi-expert system invokes the score comparison module. If the score of a subclass exceeds the Minimum-score threshold and the Contention threshold, the identity of that subclass will be recognized as the identity of the input character (right or wrong - recognition or substitution), else it will be a rejection case.

For example, let the Contention threshold be 1.2 and the Min-score threshold be 30. Then an input character with a score of 32 as an "open zero" and 64 as an "open six" will be recognized as a six because its score as a six is greater than 30 and its contention value (64/32 = 2) is greater than 1.2

The main function of the multi-expert system iterates all the above steps for each input character until it reaches the end of the input file Having the real identity of the input characters as part of the input file, the main function produces statistical reports about the recognized, substituted and rejected cases.

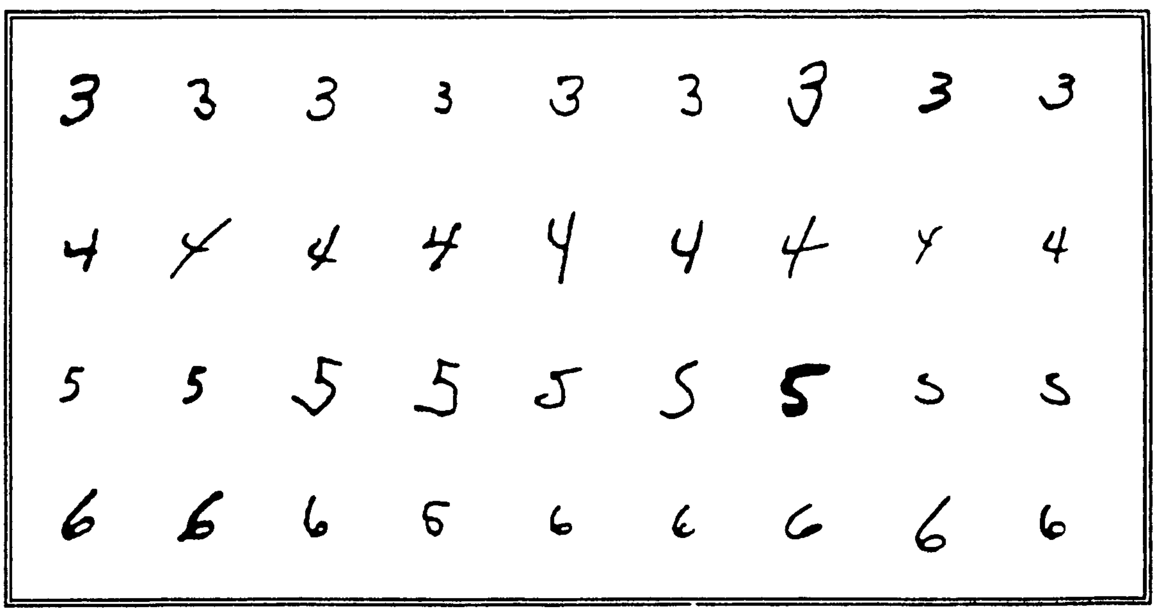


Figure 6.2: Samples of training and test data

6.3 Democratic or Non-democratic Voting

According to the formula in Section 5.3.2, the score gained by firing of a rule is directly related to its expert weight. Experts weights can vary between one and zero A rule (of a subclass) with a zero expert weight will have no effect on the total score gained for its subclass, while an expert weight of one (with a line weight of one) is needed to gain the maximum score of any rule.

One way of assigning weights to experts is to consider that all experts are equally qualified For example an expert weight of one for all experts. The other approach is to have different weights for different experts, but on what basis?

Figure 2 5 and Table 2 4 show the performances of five experts with the set of 355 samples Performances of experts can be used as the basis for determining their weights One way of arriving at a weight for an expert is to use his/her substitution rates only, as in the following formula. Though a formula that includes both recognition and substitution rates could be more versatile

$$W_e = \frac{10-S_e}{10} \qquad \{\text{if } s_e \geq 10,\ W_e = 0\}$$

Where

W_e -Weight of expert e $\quad 0 \leq W_e \leq 1$

S_e - Substitution rate of expert e

By using the above formula and the entries in Table 2.4 the following table for experts weights was generated.

Expert	Substitution Rate	Expert Weight
1	1 7	0.83
2	4.2	0.58
3	0	1
4	6.2	0.38
5	4.2	0.58
All (average)	3.3	0.67

Table 6.1: Experts' weights based on their performances.

Throughout the training and testing processes, equal expert weights (one for all experts) and unequal expert weights (Table 6.1) were both used This was done in order to examine the effects of having different weights for different experts.

6.4 Training and Testing

Almost all off-line handprint recognition systems are based upon trainable algorithms ([16]). This multi-expert system (before and after training) was tested on the set of 355 confusing cases using equal and unequal expert weights. For comparison purpose, performance of human experts ([27]), human volunteers and multi-expert system (see Section 2.1) are shown together in Table 6.2 and Figure 6.3.

	EXPERT W T.						LINE W T.						
	All Exp	E1	E2	E3	E4	E5	L1	L2	L3	L4	Rec.	Sub.	Rej.
Human Experts	-	-	-	-	-	-	-	-	-	-	59.2 %	0.0 %	40.8 %
Human Volunteers	-	-	-	-	-	-	-	-	-	-	46.4 %	6.1 %	47.5 %
Expert System (Before Training)	1	1	1	1	1	1	1	1	1	1	35.21 %	8.45 %	56.33 %
Expert System (Before Training)							1	1	1	1	36.36 %	7.10 %	56.54 %
	0 67	0 83	0 58	1	0.38	0.58	1	0.5	0	0	37.18 %	7.60 %	55.21 %
							1	0.5	.25	.25	37.74 %	6.19 %	56.07 %
Expert System After (Limited) Training with Data Sets A & B	0.67	0.83	0.58	1	0.38	0.58	1	0.5	.25	.25	46.47 %	11.83 %	41.69 %

Table 6.2. Human and E S. performance with confusing cases of numerals.

Figures 6.4, 6.5 and 6.6 show some of the samples among the 355 numerals that were recognized, substituted and rejected respectively by the multi-expert system

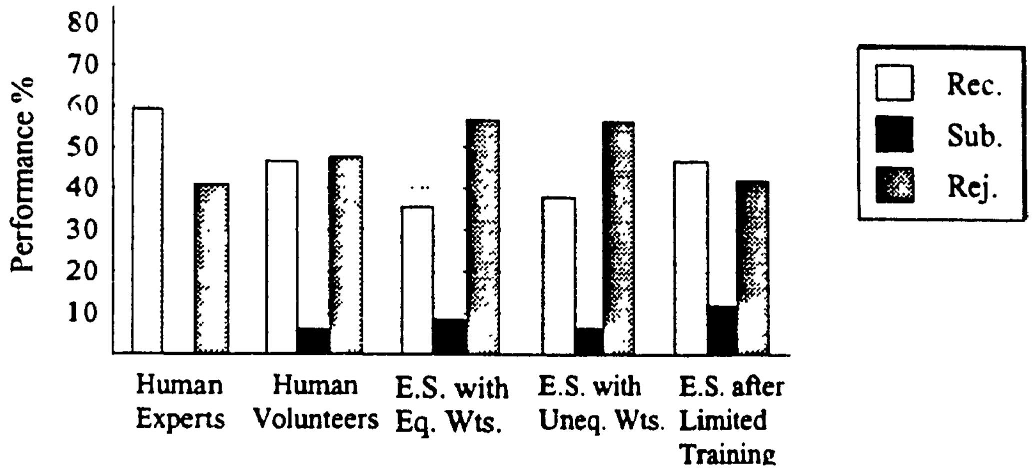


Figure 6.3: Human and E.S. performance with confusing cases of numerals

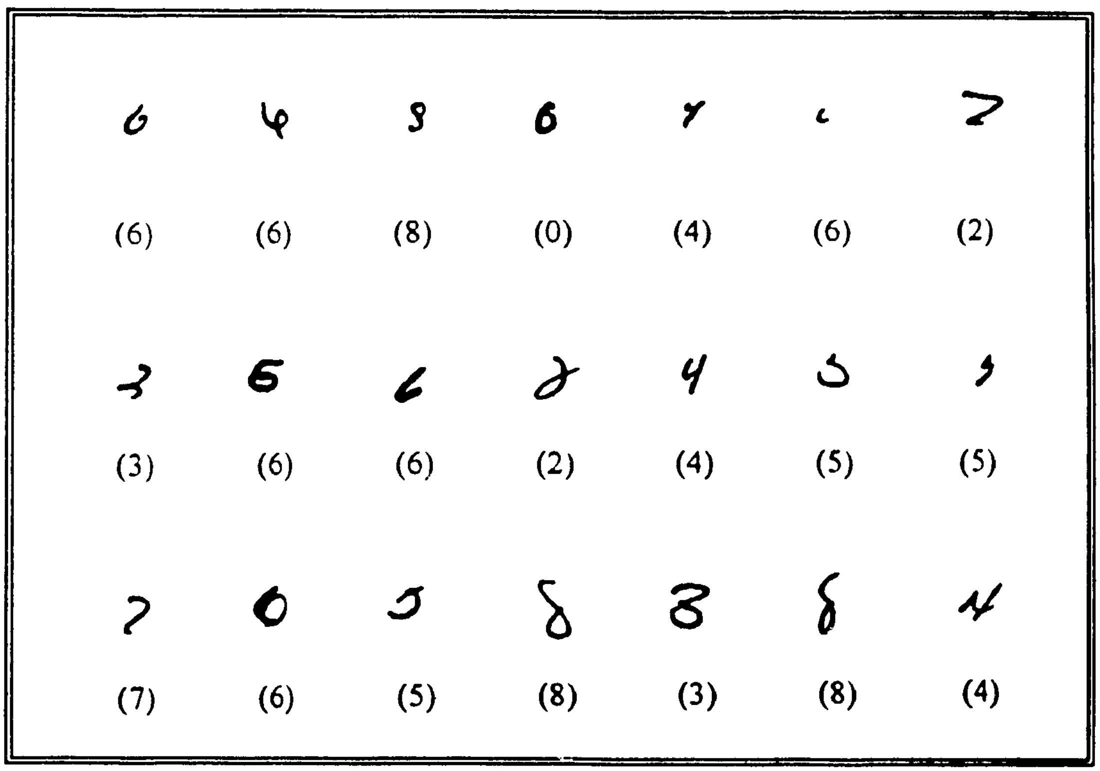


Figure 6.4: Examples of correctly recognized numerals.

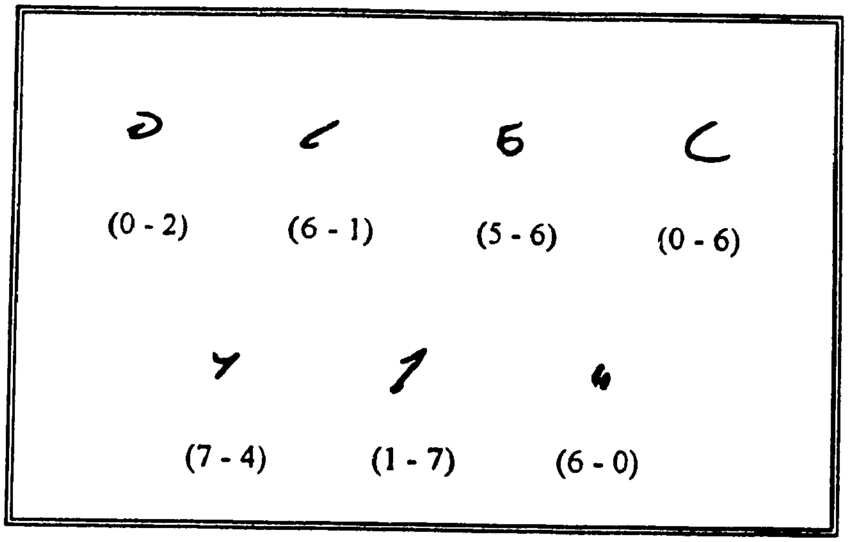


Figure 6.5: Examples of the substitution cases.
Left number inside the brackets is the real id. and the right number is the id given by the multi-expert system.

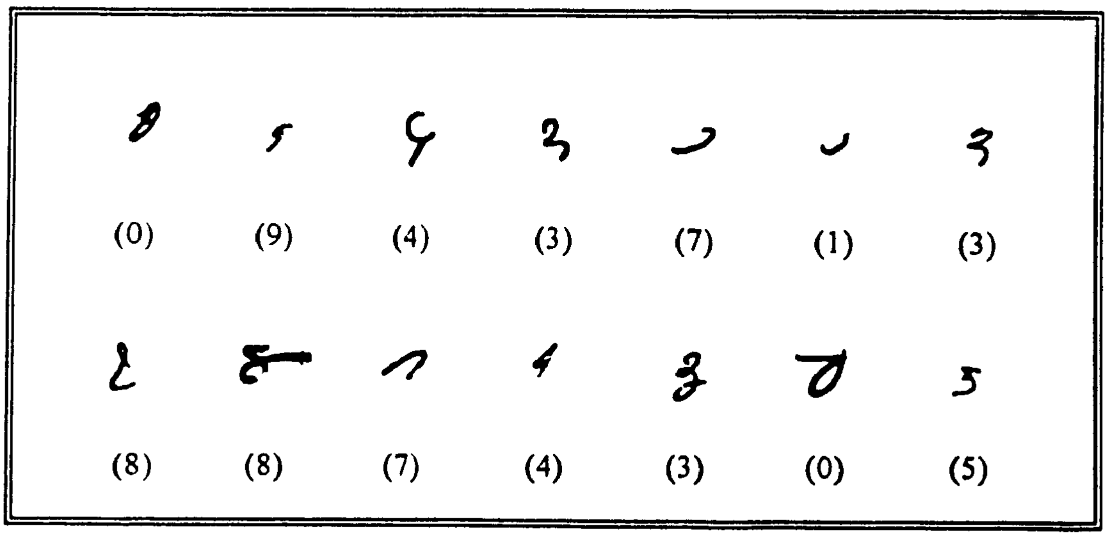


Figure 6.6: Examples of rejected characters.
Characters in the upper row are rejected because they do not pass through the threshold values, and those in the lower row are rejected because their skeletons were unknown to the multi-expert system.

The numbers in Table 6.2 indicate that human experts performed far better than the multi-expert system implemented However, performance of the multi-expert system may be compared to that of the volunteer (naive) human subjects. The overall superior performance of human subjects can be explained on the basis of the following facts.

- Knowledge base of the multi-expert system is incomplete in terms of scope and quality (see Section 2.3).
- Absence of feedback by experts makes alteration of rules (further training) impossible (see Section 3.3).
- Bias in favour of experts' judgement, since identity label of the 355 samples are based on the majority vote of experts' classification (see Section 2 4).
- Noise in the binarized input samples (see Figure 4.3)
- Relying on a single method for extracting features (see Section 4.3).

Another important observation is that the multi-expert system with unequal experts and lines weights has performed better (though slightly) than that with highest equal weights for experts and lines. This can be explained as follows,

1. When the scores of the members of a group (which is under examination) are not distant enough, the inference engine refuses to make any decision about the identity of the input sample This kind of rejection can happen when all subclasses of a group score high enough to get too close to each other and consequently not satisfying the Contention threshold.
2. With equal weights for lines and experts (here a weight of one for all of them), it becomes easier for subclasses to score enough to exceed the Minimum-score threshold. This reduces rejections

due to Minimum-score threshold and naturally adds to the recognition and substitution rates.

Before training the multi-expert system on data set 'A', it was tested with that and its results are shown in Table 6.3 and Figure 6.3. The substitution rate, despite lack of any training, has a very low value. Then the system was trained on data set 'A' which involved:

1 Refinement of implementation of rules.

2 Tuning of the parameters that define the characteristics of the primitives of a sample. For example if a line is short, medium or long. Best values for these parameters can be found automatically.

3 Refinement of Group-detector module.

4 Addition of code to the peculiar cases classifier module for the recognition of previously unknown peculiar cases.

5. Variation of expert and line weights automatically for achieving the best performance.

The results of testing the multi-expert system on data sets 'A' and 'T' after training it on data set 'A', both with equal and unequal weights, are shown in Figure 6 7 and Table 6.3.

For both sets, 'A' and 'T', the performance of the multi-expert system is better when experts and lines weights have different values. Since the number of confusing cases in these 2000 sample sets is smaller (compared to that for the set of 355 confusing cases) the difference in performance with equal and unequal weights is very slim

Again, it is noticeable that the substitution rate for both data sets 'A' and 'T', is very low. In order to provide more details, a confusion matrix of the performance of the multi-expert system on set 'T' after training it on data set 'A' with unequal weights is shown in Table 6.4.

Finally, the expert system was trained on the data set 'B' as it had been done with the data set 'A'. Then it was again tested with data set 'T'. For comparison purpose the results of testing the expert system with set 'T' before and after training it on set 'B' are shown in Table 6.5 and Figure 6.8.

The second level of training enhanced the recognition rate by 5% and slightly reduced the substitution rate. Again, the recognition rate is better when experts and lines have unequal weights, also the substitution rate has a low value. A confusion matrix of the performance of the multi-expert system on set 'T' after training it on sets 'A' and 'B' with unequal weights is shown in Table 6.6.

	EXP - Wt						L - Wt						
	Exp All	E1	E2	E3	E4	E5	L1	L2	L3	L4	Rec	Sub.	Rej
Set 'A' Before Training	0.67	0.83	0.58	1	0.38	0.58	1	0.5	0	0	61.90 %	2.20 %	35 90 %
Set 'A' After Training Equal Wt	1	1	1	1	1	1	1	1	1	1	74.90 %	1.20 %	23 90 %
Set 'A' After Training Unequal Wt.	0.67	0 83	0.58	1	0.38	0.58	1	0.5	0	0	75.60 %	0.90 %	23 50 %
Set 'T' Equal Wt.	1	1	1	1	1	1	1	1	1	1	71.30 %	1.95 %	26 75 %
Set 'T' Unequal. Wt.	0.67	0.83	0.58	1	0.38	0.58	1	0.5	0	0	71 80 %	2 0 %	26.20 %

Table 6.3. Performance of E.S. before and after training on set 'A'.

Figure 6 7. Performance of E.S. before and after training on set 'A'.

		R	E	S	P	O	N	S	E						
		0	1	2	3	4	5	6	7	8	9	Total	Rec %	Sub %	Rej %
	0	154		4			1	1				200	77	3	20
	1		197									200	98 5	0	1.5
S	2			130	2			1	2	3	1	200	65	4.5	30.5
T	3				118				1			200	59	0.5	40.5
I	4					153					1	200	76.5	0.5	23
M	5		3				125	1			6	200	60 5	5	34.5
U	6	1				1		135				200	67.5	1	31 5
L	7		2	1		3	2		132			200	66	4	30
I	8			1						158		200	79	0.5	20.5
	9			1						1	134	200	67	1	32
											Total	2,000	71.8	2	26.2

Table 6.4: Confusion matrix for set 'T' after training on set 'A'.

In all the tests more than 70% of the samples were passed through the confusing cases classifier (see Figure 1.2) and only the remaining ones were processed by the peculiar cases classifier This indicates that the limitations of the knowledge base in terms of scope or knowledge can dramatically affect the overall performance of the multi-expert system.

In all tests, substitution rate of the multi-expert system remained very low and its recognition rate improved considerably by training. Overall assessment of the system is discussed in Chapter 7.

	EXP - Wt						L - Wt						
	Exp All	E1	E2	E3	E4	E5	L1	L2	L3	L4	Rec	Sub	Rej
Set 'T' Equal Wt Trained on 'A' only	1	1	1	1	1	1	1	1	1	1	71.30 %	1 95 %	26 75 %
Set 'T' Unequal Wt trained on 'A' only	0.67	0.83	0.58	1	0 38	0.58	1	0.5	0	0	71.80 %	2 0 %	26 20 %
Set 'T' Equal Wt trained on 'A' & 'B'	1	1	1	1	1	1	1	1	1	1	76.20 %	1 85 %	21.95 %
Set 'T' Unequal. Wt. trained on 'A' & 'B'	0.67	0.83	0.58	1	0.38	0.58	1	0.5	0	0	76 80 %	1 95 %	21.25 %

Table 6.5: Performance of E.S. on set 'T' after training it on sets 'A' and 'B'

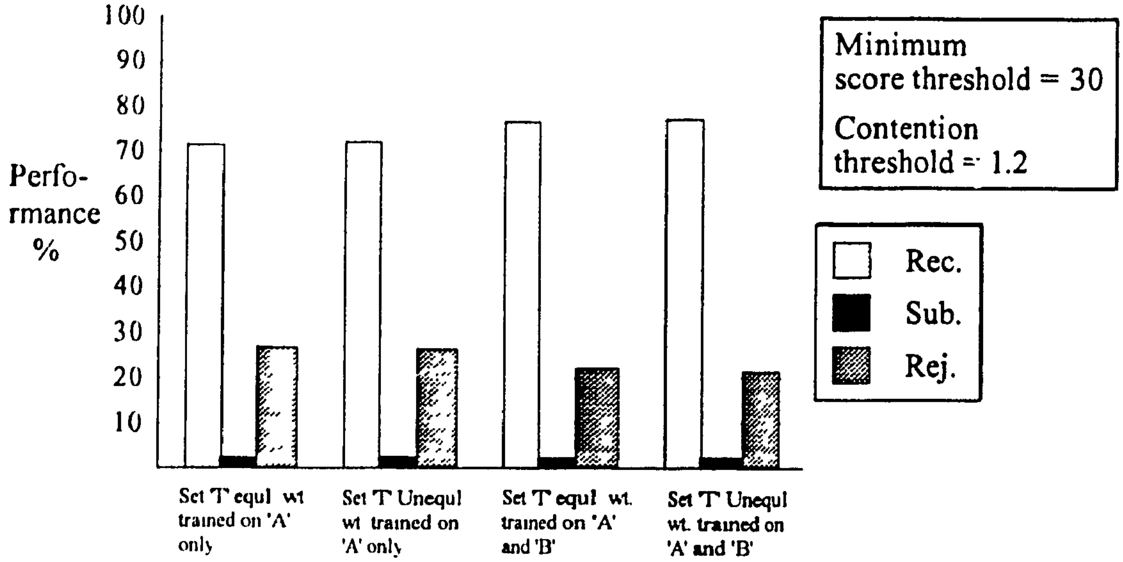


Figure 6 8 Performance of E.S. on set 'T' after training it on sets 'A' and 'B'.

		RESPONSE													
		0	1	2	3	4	5	6	7	8	9	Total	Rec. %	Sub. %	Rej. %
	0	156		5			1	1				200	78	3.5	18 5
	1		181									200	90 5	0	9.5
S	2			130	2			3	3	1	1	200	65	5	30
T	3				150				3			200	75	1.5	28
I	4					156					1	200	78	0.5	21.5
M	5		1				152	2	1		3	200	76	3 5	20 5
U	6	1						162				200	81	0.5	18.5
L	7		1	2		3	2		147			200	73.5	4	22.5
I	8			1						158		200	79	0.5	20.5
	9									1	144	200	72	0.5	27.5
											Total	2,000	76.8	1.95	21.25

Table 6.6: Confusion matrix for set 'T' after training on sets 'A' and 'B'.

Chapter 7

Conclusion

The main objective of this work was to examine the development process and performance of a recognition system that classifies confusing cases of unconstrained handwritten numerals using multiple human expertise This system was implemented as a rule based expert systems and despite serious limitations of its knowledge base promising results were obtained.

Consistent results were obtained throughout the experiment which are considered to be reliable because the data sets that were used had been validated by human experts.

According to test results this numeral recognition system maintained low substitution rates with different data sets and consistently improved on its recognition rate by further training In spite of limitations of its knowledge base (in terms of scope and quality), its performance is comparable to that of some complete recognition systems which do not use human expertise in this field as it is used in this system ([28]).

There are other interesting and important findings due to this work which are summarized as follows:

- Recognition of confusing cases of handwritten characters must be the target of new handwritten character recognition systems rather than constructing complete systems again and again.
- Classes of characters can be considered as sets of their variations (or "subclasses"). Subclasses of characters are the real source of

confusion for character recognition systems, not classes of characters

- In a multi-expert recognition system, better results are obtained when experts weights are derived from their previous performances rather than using equal weights for all experts.
- For a character recognition multi-expert system, knowledge acquisition from experts must be conducted interactively.
- Construction of a numeral recognition system based on the expertise of a number of experts needs appropriate planning ahead specially in the knowledge acquisition and elicitation phases Moreover its construction takes much longer time than that for a system which is based solely on mathematical formulas.

Performance of this multi-expert recognition system can be enhanced to a great extent if the following improvements are incorporated in it.

1. Knowledge acquisition: Experts must be given the opportunity to assign weights to features directly instead of limiting them to line weights Experts must distinguish key features from other features More than 360 confusing samples must be examined by human experts in order to build a complete recognition system. Experts and knowledge engineer(s) must be able to interact directly or indirectly
2. Knowledge elicitation. Experts' weights may be determined on the basis of their performances with subclasses instead of classes of numerals. Experts must provide feedback during the knowledge elicitation phase. Expert code of those features which are observed by more than one expert must be indicative of the participant experts.

3. Preporcessing and Feature extraction The binarized matrix of an input character must represent the actual character properly More than one method of feature extraction must be used.
4. Inference engine: Rules in the subclasses of the selected groups can be examined (fired) in parallel. The aggregation formula may be appended to include the changes which are suggested in point no. 2 above.
5. Training and Testing: Larger data sets must be used for the training and testing of the system. Training of the system needs modification of the rules as well which can be done more appropriately with feedback from experts A confusion matrix for the subclasses of numerals instead of that for the classes of numerals could pin-point the most confusing shapes of numerals more precisely.

References

[01] R D. Boyle and R. C. Thomas, Computer Vision a First Course, Blackwell Scientific Publications Ltd., Boston, 1988.

[02] T. A Mai, "A Knowledge-Based Approach to Recognize Unconstrained Handwritten Numerals," M Sc. Thesis, Concordia University, Montreal, Canada, 1989.

[03] H Goraine, M. Usher and S. Al-Emami, "Off-Line Arabic Character Recognition," IEEE Computer, vol. 25, No. 7, pp. 71-74, July 1992.

[04] O Matan, H. S. Baird et al, "Reading Handwritten Digits: A ZIP Code Recognition System," IEEE Computer, vol 25, No. 7, pp. 59-62, July 1992

[05] C C Tappert, C. Y. Suen and T. Wakahara, "The State of the Art in On-Line Handwriting Recognition," IEEE Trans. Patt. Anal. and Machine Intell., vol. 12, No. 8, pp. 787-808, Aug. 1990.

[06] K Parsaye and M Chignell, Expert Systems for Experts, John Wiley and Sons, Inc. , Toronto, 1988.

[07] D A Waterman, A Guide to Expert Systems, Addison-Wesley Publishing Company, Inc., Toronto, 1985.

[08] F. Ali and T. Pavlidis, "Syntactic Recognition of Handwritten Numerals," IEEE Trans. Syst., Man, Cyber., vol. SMC-7, No. 7, pp.537-541, July 1977.

[09] J S. Huang and K. Chuang, "Heuristic Approach to Handwritten Numeral Recognition," Pattern Recognition, Vol. 19, No. 1, pp. 15-19, 1986.

[10] E C. Schwab and H. C. Nusbaum, Pattern Recognition by Humans and Machine, Vol. 2, Academic Press Inc., New York, 1986.

[11] C. Y. Suen (ed.), Frontiers in Handwriting Recognition, CENPARMI, Concordia University, Montreal, Canada, 1990.

[12] U. Neisser and P. Weene, "A Note on Human Recognition of Handprinted Characters", Information and Control, vol. 3, 191-196, 1960.

[13] R. Legault, C. Y. Suen and C. Nadal., "Classification of Confusing Handwritten Numerals by Human Subjects," in Frontiers in Hanwriting Recognition, CENPARMI, Concordia University, Montreal, Canada, pp 181-193, 1990.

[14] A S. Elmaghraby, "A Prolog Simulation for a Delphi-Based Problem Solver," Simulation Digest, acm Press, Vol 19, No. 4, pp. 36-43, 1989

[15] G. S. Tuthill, Knowlege Engineering; Concepts and Practices for Knowlege-Based Systems, TPR Press, Blue Ridge Summit PA, 1990

[16] C. Y. Suen, M. Berthod and S. Mori, "Automatic Recognition of Handprinted Charcters - The State of the Art," Proc. of the IEEE, vol 68, pp 469-487, Apr. 1980.

[17] C. Y. Suen, C. Nadal, T. A. Mai, R. Legault and L. Lam, "Recognition of Totally Unconstrained Handwritten Numerals Based on the Concept of Multiple Experts", in Frontiers in Handwriting Recognition, CENPARMI, Concordia University, Montreal, Canada, pp 131-143, 1990.

[18] C. Y. Suen and R. J. Shillman, "Low Error Rate Optical Character Recognition of Unconstrained Letter Based on a Model of Human Perception", IEEE Trans. Syst., Man, Cyber., vol. 7, 491-495, 1977.

[19] L Lam, "Structural Classification and Relaxation Matching of Totally Unconstrained Handwritten Numerals," M.Sc. Thesis, Concordia University, Montreal, Canada, 1986.

[20] Waiqian Dai, "Recognition of Handwritten Numerals Using Neural Networks," M Sc. Thesis, Concordia University, Montreal, Canada, 1990.

[21] H. Genchi, K . Mori et al, "Recognition of Handwritten Numerical Characters for Automatic Letter Sorting," Proc. of the IEEE, vol. 56, No. 8, pp. 1292-1301, 1968.

[22] J T. Tou and R C. Gonzalez, "Recocgnition of Handwritten Characters by Topological Featu-e Extraction and Multilevel Categorization," IEEE Trans on Computers, Vol. C-21, No. 7, pp. 776-785, 1972.

[23] P Ahmed and C. Y. Suen, "Computer Recognition of Totally Unconstrained Handwritten ZIP Codes," Inter. Jour. of Patt. Recog. and Art. Intell. , vol. 1, No. 1, pp 1-15, 1987.

[24] J R Ward, "One View of On-Going Problems in Handwriting Character Recognition," Proc. Workshop at CENPARMI, Concordia University, Montreal, Canada, pp. 101-106, 1991.

[25] A. Kandel, Fuzzy Techniques in Pattern Recognition; John Wiley & Sons, New York, 1982.

[26] R Legault and C Y. Suen, "A Contour-Based Recognition System for Totally Unconstrained Handwritten Numerals," Technical Report, CENPARMI, Concordia University, Montreal, Canada, July 1989.

[27] R. Legault, C. Y. Suen and C. Nadal, "Difficult Cases in Handwritten Numeral Recognition," Technical Report, CENPARMI, Concordia University, Montreal, Canada, July 1991.

[28] C. Nadal and C. Y. Suen, "Recognition of Unconstrained Handwritten Digits by Decomposition and Vectorization," Tech Report, CENPARMI, Concordia University, Montreal, Canada, 1988.

[29] J. Boose and B. Gaines, Knowledge Acquisition Tools for Expert Systems, Academic Press, New York, 1991.

[30] A. Hart, Knowledge Acquisition for Expert Systems, McGraw-Hill Book Company, New York, 1986.

[31] V. Jagannathan and A. S Elmaghraby, "MEKDAT: Multiple Expert Delphi-Based Knowledge Acquisition Tool", Proc. of ACM, Second Annual acm Northeast Regional Conference , pp 30-34, Oct 1985

[32] I. Graham and P. L. Jones, Expert Systems; Knowledge, Uncertainty and decision, Chapman and Hall, New York, 1988.

[33] G. A. Mack, J. F. Brule and J Levin, "A Cost-Effective, Repeatable Method for Managing the Development of Multiple-Expert Expert Systems," Proc. of IEEE Conference on Managing Expert System Programs and Projects, pp. 125-133, Sep. 1990.

[34] K. L. Mcgraw and k Harbison - Briggs, "Acquiring Knowledge from Multiple Experts," Knowledge Acquisition: Principles and Guidelines, pp 240-271, Prentice - Hall, 1989.

[35] B Brown, "Pen Computing: Operating Systems, Hardware, and Apps," Byte Magazine, Vol. 11 No. 12, pp. 37, June 30 1992.

[36] M. Meyer and J. Booler, Eliciting and Analyzing Expert Judgement; A Practical Guide, Academic Press, New York, 1991

[37] M. Hu, Y. Wang and T. Qin, "Multiexpert Opinion Combination Mechanisms and Their Knowledge Organizations," IFAC Man-Machine Systems, Xi'an, PRC, 1989.

[38] L. Lam, S. Lee and C. Y. Suen, "Thinning Methodologies - A Comprehensive Survey," IEEE Trans. Patt. Anal. and Machine Intell., Vol. 14, pp. 869-885, 1992.

[39] A. D. Preece, C. Y. Suen and C. L. Yu, "Performance Assessment of a Charcter Recognition Expert System," I.I.T.T. Technology Transfer Series, pp. 295-300, 1990.

[40] B. Blesser, R. Shillman, T. Kuklinski, C. Cox, M Eden, and J. Ventura, "A theoretical approach for character recognition based on phenomenological attributes," International Journal of Man-Machine Studies, vol. 6, pp. 701-714, 1974.

[41] C. Y. Suen, "Distinctive Features in Automatic Recognition of Handprinted Characters," Signal Processing, Vol. 4, pp. 193-207, North-Holland Publishing Company, Holland, 1982.

[42] A. G. Du Boisdulier and P. Vanhulle, "Topological Modelization of Handwritten Digits Based on a Geometrical Primitive Decomposition," CGA-HBS, France, 1989.