# Style Classification of Cursive Script Recognition

A thesis submitted in partial fulfilment of the requirements of the Nottingham Trent University for the degree of Doctor of Philosophy


***Mandana Ebadian Dehkordi***


May 2003


*Department of Computing and Mathematics*
*The Nottingham Trent University*
*Burton Street, Nottingham, NG1 4BU,UK*

*mandana.dehkordi@medicsight.com*

*Tel: +44-115-8484742;Fax: +44-115-9486518*

# STYLE CLASSIFICATION OF CURSIVE SCRIPT RECOGNITION


## ABSTRACT

Handwriting recognition has been the subject of intensive research for many years. However, despite the best effort of many researchers, the problem of handwriting recognition is far from being solved. The greatest difficulty in cursive script recognition is due to the large variation of shapes that can result from the different writing styles. A common way to address this problem is to accommodate the variability in the feature set. However, such systems are limited in the range of writing styles that they can successfully deal with. An alternative approach has been to minimise the variability within the handwriting itself. Techniques such as normalisation, slant correction, restricting the number of objects to be recognised (i.e. numerical character, name of city) etc. have been shown to be partially effective. However further work remains to be done in order to cope with the variation problem. Here it is hypothesised that a pre-classification of writer style would provide an effective means of managing style variation and hence achieve better recognition results.

The main aim of this thesis is to investigate alternative ways of addressing problems brought about by the variability of human handwriting; in particular those problems related to the recognition of off-line cursive handwriting. Style has been further broken down into case and quality of handwriting. Case classification of handwriting is proposed as a means of limiting the size of the template database used for word recognition. The quality of handwriting has been defined in terms of its legibility. It is proposed that this approach would lead to determining the legibility of an unknown sample prior to recognition. So as to select a recogniser that is suited to the quality of handwriting of the unknown sample.

Two non-parametric classification techniques are applied to features extracted from the word image contours in order to compare their effectiveness in classifying words into upper, lower and mixed cases and further into legible, illegible and middle (between legible and illegible) classes. In the first method, a Multiple Discriminant Analysis (MDA) is used to transform the space of the extracted feature (36 dimensions) into an optimal discriminate space for a nearest mean based classifier. In the second method, a Probabilistic Neural Network (PNN) based on Bayes strategy and non-parametric estimation of probability density function is used. The experimental results show that PNN gives superior classification results when compared to MDA for both types of style classification.

A number of experiments have been carried out using unseen data to determine the effectiveness of the above techniques. For a two-class word case classification problem the PNN approach yields 100% (lower/upper), 88%(upper/mixed) and 81%(lower/mixed) correct classification. For three-class word case classification the rate of correct classification is 73%. The same approach when applied to legible, illegible and middle style classification handwriting provides 86.5% (legible/illegible), 75.5% (legible/middle) and 90.5% (middle/illegible) correct classification for two classes. For three-class legibility classification the rate of correct classification is 67.33%.

Style variation remains an open subject for further research. Word case and legibility are demonstrated to provide positive steps towards a more tangible definition of style. This research has demonstrated that a holistic classification technique is effective in dealing with the concept of style in a quantifiable manner. The experimental results indicate that further word level features are needed to further improve classification. This together with additional style categories would lead to more effective means of managing variability.

*The work describe in this thesis is the author's own unless stated otherwise.*

*It contains - to the author's knowledge-original material.*

# ACKNOWLEDGEMENTS

The author is indebted to many friends and colleagues for help and encouragement- too many for all to be mentioned by name. There are a few who must be named, however, because without their assistance the completion of this project would have not been possible.

In particular I would like to thank Dr. Nasser Sherkat and Dr. Tony Allen for their guidance, encouragement, valuable comments and numerous interesting discussions throughout this work.

I would like to thank professor Mike Fairhurst as my External exam and Dr. Joanna Hartley as my internal exam for their helpful comment and interesting discussions throughout my viva.

I would also like to express my appreciation to my husband Dr. Jamshid Dehmeshki who helped make this work possible.

Finally, I must thank my father, Mr. Ibrahim Ebadian Dehkordi, mother, sister, brothers and my son for their love and moral support and my other colleagues in Nottingham Trent University.

# Table of Contents

**Chapter 1.** Introduction ........ 1-1

1.1 Machine printed writing style characterisation ........ 1-1

1.2 Cursive script style characterisation ........ 1-2

1.3 The objectives of the project ........ 1-6

1.4 Outlines of the thesis ........ 1-9

**Chapter 2.** Literature review ........ 2-1

2.1 Optical character recognition ........ 2-2

2.2 Properties of handwriting ........ 2-5

2.3 Handwriting recognition techniques ........ 2-6

2.4 Style classification and recognition improvement ........ 2-9

2.5 Future direction ........ 2-16

2.6 Summary ........ 2-17

**Chapter 3.** Feature extraction ........ 3-1

3.1 Introduction ........ 3-1

3.2 Contour-based features ........ 3-1

3.2.1 Global features ........ 3-4

3.2.2 Region-based features ........ 3-6

3.2.3 Windows-based features ........ 3-11

3.2.4 Features-based moments ........ 3-13

3.2.5 Zero crossing feature ........ 3-14

3.3 Group-based features ........ 3-15

3.3.1 Group Definition ........ 3-15

3.3.2 Group-based feature ........ 3-17

3.4 Horizontal-based histogram features ........ 3-18

3.5 Summary ........ 3-21

**Chapter 4.** Classification methods ........ 4-1

4.1 Introduction ........ 4-1

4.2 Linear classification-Multiple Discriminant Analysis method ........ 4-2

4.3 Non-linear classification PNN method ........ 4-5

4.3.1 Parzen method ........ 4-6

4.3.2 Optimizing the $\sigma$ ........ 4-7

4.3.3 Probabilistic Neural Network ........ 4-8

4.3.4 Comparison of appropriate classification methods .............................. 4-10
4.4 Feature efficiency .............................................................................................. 4-11
4.4.1 Feature efficiency in case classification ................................................ 4-13
4.4.2 Feature efficiency for handwriting legibility classification .................. 4-17
4.5 Summary ........................................................................................................... 4-21
**Chapter 5.** Case classification ................................................................................ 5-1
5.1 Introduction ....................................................................................................... 5-1
5.1.1 Handwriting data samples ........................................................................ 5-1
5.2 PNN using common $\sigma$ .................................................................................... 5-4
5.2.1 Binary classification ................................................................................... 5-4
5.2.1.1 Experimental results and analysis using 36 extracted features ..... ....................................................................................................... 5-4
5.2.1.2 Experimental results and analysis using the selected features ...... ....................................................................................................... 5-7
5.2.1.3 Comparison between the selected and 36 extracted features ..... ....................................................................................................... 5-10
5.2.2 Triple classification ................................................................................... 5-12
5.2.2.1 Experimental results and analysis using 36 extracted features ..... ....................................................................................................... 5-12
5.2.2.2 Experimental results and analysis using the selected features ...... ....................................................................................................... 5-14
5.2.2.3 Comparison between using the selected and 36 extracted features ............................................................................................ 5-15
5.3 PNN using different $\sigma_i$ .................................................................................. 5-17
5.3.1 Binary classification ................................................................................... 5-17
5.3.1.1 Experimental results and analysis using 36 extracted features ..... ....................................................................................................... 5-18
5.3.1.2 Experimental results and analysis using selected features ..... 5-20
5.3.1.3 Comparison between using the selected and 36 extracted features ............................................................................................ 5-22
5.3.2 Triple classification ................................................................................... 5-23
5.3.2.1 Experimental results and analysis using 36 extracted features ..... ....................................................................................................... 5-24

5.3.2.2 Experimental results and analysis using the selected features ...... ........................................................................ 5-25
5.3.2.3 Comparison between using the selected and 36 extracted features ........................................................................ 5-27
5.4 Comparison between the selected and 36 extracted features in triple classifications using PNN method ........................................ 5-29
5.4.1 36 extracted features ........................................................ 5-29
5.4.2 Selected features ............................................................ 5-30
5.5 Multiple linear classification (MDA) ........................................ 5-30
5.6 Comparison between the linear and non-linear methods for binary case classification ........................................................................ 5-33
5.6.1 36 extracted features ........................................................ 5-34
5.6.2 Selected features ............................................................ 5-34
5.7 Conclusion ........................................................................ 5-35
**Chapter 6.** Prediction of legibility based on existing recogniser ............ 6-1
6.1 Introduction ........................................................................ 6-1
6.2 Definition of legibility ........................................................ 6-2
6.3 PNN style classification using common $\sigma$ ................................ 6-4
6.3.1 Binary classification ........................................................ 6-4
6.3.1.1 Experimental results and analysis using 36 extracted features....... ........................................................................ 6-4
6.3.1.2 Experimental results and analysis using the selected features ....... ........................................................................ 6-7
6.3.1.3 Comparison between using the selected and 36 extracted features ........................................................................ 6-10
6.3.2 Triple classification ........................................................ 6-12
6.3.2.1 Experimental results and analysis using 36 extracted features....... ........................................................................ 6-12
6.3.2.2 Experimental results and analysis using the selected features ....... ........................................................................ 6-14
6.3.2.3 Comparison between using the selected and 36 extracted features ........................................................................ 6-16
6.4 PNN style classifier using different $\sigma_i$ .................................... 6-17

6.4.1 Binary classification ........................................................................ 6-17
6.4.1.1 Experimental results and analysis using 36 extracted features....... ........................................................................................................ 6-18
6.4.1.2 Experimental results and analysis using the selected features ....... ........................................................................................................ 6-21
6.4.1.3 Comparison between using the selected and 36 extracted features ........................................................................................ 6-23
6.4.2 Triple classification ........................................................................ 6-24
6.4.2.1 Experimental results and analysis using 36 extracted features....... ........................................................................................................ 6-25
6.4.2.2 Experimental results and analysis using the selected features ....... ........................................................................................................ 6-27
6.4.2.3 Comparison between using the selected and 36 extracted features........................................................................................ 6-28
6.5 Comparison between using the selected and 36 extracted features in triple classification using PNN method ................................................ 6-30
6.5.1 36 extracted features ........................................................................ 6-30
6.5.2 Selected features ........................................................................ 6-31
6.6 Multiple linear classification (MDA) ................................................ 6-31
6.7 Comparison between using the linear and non-linear method for binary case classification ........................................................................ 6-34
6.7.1 36 extracted features ........................................................................ 6-35
6.7.2 Selected features ........................................................................ 6-35
6.8 Conclusion ........................................................................ 6-36
**Chapter 7.** Conclusion and future work ................................................ 7-1
7.1 Achievements in case classification of handwriting ............................ 7-3
7.2 Achievements in handwriting legibility classification ......................... 7-4
7.3 Summary ........................................................................ 7-5
7.4 Future work ........................................................................ 7-6
**References** ........................................................................ R-1
**Appendix A.** Some samples from the database ..................................... A-1
**Appendix B.** PNN method in case classification .................................. B-1
B.1 Binary classification using common $\sigma$ ........................................ B-1
B.1.1 Analysis of zoning information using 36 extracted features .......... B-1

B.1.2 Analysis of zoning information using the selected features ........... B-4
B.2 Triple classification using common $\sigma$ ........................................................ B-6
B.2.1 Analysis of zoning information using 36 extracted features .......... B-6
B.2.2 Analysis of zoning information using the selected features ........... B-8
B.3 Binary classification using different $\sigma_i$ .................................................. B-10
B.3.1 Analysis of zoning information using 36 extracted features ........ B-10
B.3.2 Analysis of zoning information using the selected features ......... B-13
B.4 Triple classification using different $\sigma_i$ .................................................. B-15
B.4.1 Analysis of zoning information using 36 extracted features ........ B-15
B.4.2 Analysis of zoning information using the selected features ......... B-16
**Appendix C.**
Classification of Off-line Hand-written Words into Upper and Lower Cases... ... ........................................................................................................ C-1
A Principal Component Approach to Classification of Hand-written Words... ........................................................................................................ C-5
Case Classification of Off-line Hand-written Words Prior To Recognition...... ........................................................................................................ C-9
Prediction of Handwriting Legibility............................................................ C-18

# List of Tables


## Chapter 2

**Table 2-1** *Word recognition result on the test set. R(n): correct answers in a list of length n. [Gilloux94]* .......... 2-14

## Chapter 3

**Table 3-1** $a_l$ *as a function of* $(d_{l-1} - d_l)$ .......... 3-3

**Table 3-2** *Thirty-six extracted feature* .......... 3-20

## Chapter 4

**Table 4-1** *Effectiveness of each feature in classification between each pair of classes and three case classifications* .......... 4-15

**Table 4-2** *Selected features for classification between upper, lower and mixed case words images* .......... 4-16

**Table 4-3** *Effectiveness of each feature in classification between each pair of classes and three in legibility of handwriting* .......... 4-18

**Table 4-4** *Selected features for classification between, legible, illegible and Middle handwriting* .......... 4-21

## Chapter 5

**Table 5-1** *A set of significant character sequences (character # represents a word boundary), Adapted from [KASSEL95]* .......... 5-2

**Table 5-2** *A set of sentences covering significant letter sequences* .......... 5-3

**Table 5-3** *Classification results using all 36 extracted features to discriminate between lower and uppercase word images using common* $\sigma$ ($\sigma$ =34.06559) .......... 5-6

**Table 5-4** *Classification result using all 36 extracted features to discriminate between mixed and uppercase word images using common* $\sigma$ ($\sigma$ =32.31938) .......... 5-6

**Table 5-5** *Classification results using all 36 extracted features to discriminate between mixed and lower case word images using common* $\sigma$ ($\sigma$ =43.74458) .......... 5-7

**Table 5-6** *Classification result using 20 selected features to discriminate between lower and uppercase word images using common $\sigma$ ($\sigma$ =10.00000)* ..... 5-9

**Table 5-7** *Classification result using 15 selected features to discriminate between mixed and uppercase word images using common $\sigma$ ($\sigma$ =10.67619)* ..... 5-9

**Table 5-8** *Classification results using 23 selected features to discriminate between mixed and lowercase word images using common $\sigma$ ($\sigma$ =18.32981)* ..... 5-10

**Table 5-9** *Classification results using all 36 extracted features to discriminate between mixed and lower and uppercase word images using common $\sigma$ ($\sigma$ =38.97138)* ..... 5-13

**Table 5-10** *Classification results using 25 selected features to discriminate between lower, upper and mixed case word images using common $\sigma$ ($\sigma$ =22.30066)* ..... 5-15

**Table 5-11** *Thirty-six different $\sigma_i$ for each lower/upper, mixed/lower and lower/mixed case classes using the 36 extracted features* ..... 5-18

**Table 5-12** *Classification results using all 36 extracted features to discriminate between lower and uppercase words using different $\sigma_i$* ..... 5-19

**Table 5-13** *Classification results using all 36 extracted features to discriminate between mixed and uppercase words using different $\sigma_i$* ..... 5-19

**Table 5-14** *Classification results using all 36 extracted features to discriminate between mixed and lowercase words using different* ..... 5-19

**Table 5-15** *Different $\sigma_i$ for each lower/upper, mixed/lower and lower/mixed case classes using the selected feature* ..... 5-20

**Table 5-16** *Classification results using the 20 selected features to discriminate between upper and lowercase word using different $\sigma_i$* ..... 5-21

**Table 5-17** *Classification results using the 15 selected features to discriminate between mixed and upper case word using different $\sigma_i$* ..... 5-21

**Table 5-18** *Classification results using the 23 selected features to discriminate between mixed and lowercase word using different $\sigma_i$* ..... 5-21

**Table 5-19** *Thirty-six different $\sigma_i$ for each lower, upper and mixed case class using 36 extracted features* ..... 5-24

**Table 5-20** *Classification results using 36 extracted features to discriminate between lower, upper and mixed case word using different $\sigma_i$* ..... 5-25

**Table 5-21** *Twenty-five different $\sigma_i$ for lower, upper and mixed case class using the selected features* ........ 5-26

**Table 5-22** *Three case classification results using the 25 selected features to discriminate between lower, upper and mixed case word using different $\sigma_i$* ........ 5-26

**Table 5-23** *Comparison between using different $\sigma_i$ and common $\sigma$ with the selected and 36 extracted features when the training set is different with the test set* ........ 5-29

**Table 5-24** *Comparison between using different $\sigma_i$ and common $\sigma$ with the selected and 36 extracted features when the training set the same as the test set* ........ 5-29

**Table 5-25** *Classification results using all 36 extracted features to discriminate between lower and upper case word images with MDA technique* ........ 5-31

**Table 5-26** *Classification results using all 36 extracted features to discriminate between mixed and upper case word images with MDA technique* ........ 5-31

**Table 5-27** *Classification results using all 36 extracted features to discriminate between mixed and lower case word images with MDA technique* ........ 5-31

**Table 5-28** *Classification results using 20 selected features to discriminate between upper and lower case word images with MDA technique* ........ 5-32

**Table 5-29** *Classification results using 15 selected features to discriminate between mixed and upper case word images with MDA technique* ........ 5-32

**Table 5-30** *Classification results using 23 selected features to discriminate between mixed and lower case word images with MDA technique* ........ 5-32

**Table 5-31** *Comparison between the classification results when (i) PNN with using different $\sigma_i$, (ii) PNN using common $\sigma$ and (iii) MDA techniques, when the training set is the same as the test set* ........ 5-33

**Table 5-32** *Comparison between the classification results when (i) PNN with using different $\sigma_i$, (ii) PNN using common $\sigma$ and (iii) MDA technique, when the training set is different with the test set* ........ 5-33

## *Chapter 6*

**Table 6-1** *Classification results using 36 extracted features to discriminate between legible and illegible handwriting using common $\sigma$ ($\sigma$ =5.47436)* ........ 6-6

**Table 6-2** *Classification result using 36 extracted features to discriminate between legible and middle handwriting using common $\sigma$ ($\sigma$ =7.11064)* ........ 6-6

**Table 6-3** *Classification result using 36 extracted features to discriminate between middle and illegible handwriting using common* $\sigma$ ($\sigma$ =0.01386)........ 6-7

**Table 6-4** *Classification result using 16 selected features to discriminate between legible and illegible handwriting using common* $\sigma$ ($\sigma$ =0.00066)........ 6-9

**Table 6-5** *Classification result using 15 selected features to discriminate between legible and middle handwriting using common* $\sigma$ ($\sigma$ =0.0015)........ 6-9

**Table 6-6** *Classification result using 13 selected features to discriminate between illegible and middle handwriting using common* $\sigma$ ($\sigma$ =0.0001)........ 6-10

**Table 6-7** *Classification result using 36 extracted features to discriminate between legible, illegible and middle handwriting using common* $\sigma$ ($\sigma$ =0.001)........ 6-13

**Table 6-8** *Classification result using the 20 selected features to discriminate between legible, illegible and middle handwriting using common* $\sigma$ ($\sigma$ =22.22464)........ 6-15

**Table 6-9** *Thirty-six different* $\sigma_i$ *for each legible, illegible and middle class using 36 extracted features*........ 6-18

**Table 6-10** *Classification result using the 36 extracted features to discriminate between illegible and legible handwriting using different* $\sigma_i$ ........ 6-19

**Table 6-11** *Classification result using 36 extracted features to discriminate between middle and legible handwriting using different* $\sigma_i$ ........ 6-19

**Table 6-12** *Classification result using 36 extracted features to discriminate between middle and illegible handwriting using different* $\sigma_i$ ........ 6-20

**Table 6-13** *Different* $\sigma_i$ *for each legible, illegible and middle class using the selected features*.... 6-21

**Table 6-14** *Classification result using 16 extracted features to discriminate between illegible and legible handwriting using different* $\sigma_i$ ........ 6-22

**Table 6-15** *Classification result using 15 extracted features to discriminate between middle and legible handwriting using different* $\sigma_i$ ........ 6-22

**Table 6-16** *Classification result using 13 extracted features to discriminate between middle and illegible handwriting using different*........ 6-22

**Table 6-17** *Thirty-six different* $\sigma_i$ *values for each legible, illegible and middle classification using the 36 extracted features*........ 6-25

**Table 6-18** *Classification result using 36 extracted features to discriminate between legible, illegible and middle handwriting using different* $\sigma_i$ ........ 6-26

**Table 6-19** *Twenty different $\sigma_i$ for each legible, illegible and middle class using selected features* ....... 6-27

**Table 6-20** *Classification result using 20 selected features to discriminate between legible, illegible and middle handwriting using different $\sigma_i$* ....... 6-28

**Table 6-21** *Comparison between using different $\sigma_i$ and common $\sigma$ with 20 selected and 36 extracted features when the training set is the same as the test set* ....... 6-30

**Table 6-22** *Comparison between using different $\sigma_i$ and common $\sigma$ with 20 selected and 36 extracted features when the training set is different to the test set* ....... 6-30

**Table 6-23** *Classification result using 36 extracted features to discriminate between legible and illegible* ....... 6-32

**Table 6-24** *Classification result using 36 extracted features to discriminate between legible and middle* ....... 6-32

**Table 6-25** *Classification result using 36 extracted features to discriminate between middle and illegible* ....... 6-32

**Table 6-26** *Classification result using 16 selected features to discriminate between legible and illegible* ....... 6-33

**Table 6-27** *Classification result using 15 extracted features to discriminate between legible and middle* ....... 6-33

**Table 6-28** *Classification result using 13 extracted features to discriminate between middle and illegible* ....... 6-33

**Table 6-29** *Comparison between the classification results when (i) PNN using different $\sigma_i$, (ii) PNN using common $\sigma$ and (iii) MDA techniques when the training set is the same as the test set* ....... 6-34

**Table 6-30** *Comparison between the classification results when (i) PNN using different $\sigma_i$, (ii) PNN using common $\sigma$ and (iii) MDA techniques when the training set is different with the test set* ....... 6-34

# List of figures


## *Chapter 1*

***Figure 1-1*** *Samples of different writing styles* ........ *1-3*

## *Chapter 2*

***Figure 2-1*** *Types of handwriting as defined by Tapper [TAPPER84]* ........ *2-10*

## *Chapter 3*

***Figure 3-1*** *Eight primitive directions* ........ *3-2*

***Figure 3-2*** *Angle $a_l$ at point $p_l$* ........ *3-3*

***Figure 3-3*** *A typical word* ........ *3-7*

***Figure 3-4*** *Concave regions* ........ *3-7*

***Figure 3-5*** *Convex regions* ........ *3-7*

***Figure 3-6*** *Plain regions* ........ *3-7*

***Figure 3-7*** *The detected dominant points on words* ........ *3-8*

***Figure 3-8*** *Three regions of interest within a window for some different word case samples* ........ *3-11*

***Figure 3-9*** *Three region of interest within a window for some different style of handwriting (one specific word)* ........ *3-11*

***Figure 3-10*** *Representation of the four directions (slopes)* ........ *3-12*

***Figure 3-11*** *Horizontal lines are drawn from the centre of each word* ........ *3-15*

***Figure 3-12*** *Upper and lower zone* ........ *3-16*

***Figure 3-13*** *Groups and their bounding box* ........ *3-16*

***Figure 3-14*** *Illustration of group-based features* ........ *3-18*

***Figure 3-15*** *Horizontal histogram for lower case word* ........ *3-19*

***Figure 3-16*** *Horizontal histogram for uppercase word* ........ *3-19*

***Figure 3-17*** *Horizontal histogram for mixed case word* ........ *3-19*

***Figure 3-18*** *Horizontal histogram for different style of writing* ........ *3-19*

## *Chapter 4*

***Figure 4-1*** *Organization for classification of pattern into categories* ........ *4-9*

**Figure 4-2** *The 20 largest eigenvector weights capture (93%) of the variability between the lower and upper case word images* ........ 4-15

**Figure 4-3** *The 15 largest eigenvector weights capture (94%) of the variability between the upper and mixed case word images* ........ 4-15

**Figure 4-4** *The 23 largest eigenvector weights capture (98%) of the variability between the lower and mixed case word images* ........ 4-15

**Figure 4-5** *The 25 largest eigenvector weights capture (98%) of the variability between the lower, upper and mixed case word images* ........ 4-16

**Figure 4-6** *The 20 largest eigenvector weights capture (97%) of the variability between, legible, illegible and middle handwriting* ........ 4-19

**Figure 4-7** *The 16 largest eigenvector weights capture (95%) of the variability between, legible and illegible handwriting* ........ 4-19

**Figure 4-8** *The 15 largest eigenvector weights capture (93%) of the variability between, legible and middle handwriting* ........ 4-20

**Figure 4-9** *The 13 largest eigenvector weights capture (96%) of the variability between middle and illegible handwriting* ........ 4-20

## Chapter 5

**Figure 5-1** *Error estimation of common $\sigma$ for a classification between lower and upper case word images using 36 extracted features ($\sigma$ =34.06559)* ........ 5-5

**Figure 5-2** *Error estimation of common $\sigma$ for a classification between upper and mixed case word images using 36 extracted features ($\sigma$ =32.31938)* ........ 5-5

**Figure 5-3** *Error estimation of common $\sigma$ for a classification between lower and mixed case word images using 36 extracted features ($\sigma$ =43.74458)* ........ 5-6

**Figure 5-4** *Error estimation of common $\sigma$ for a classification between lower and upper case word images using 20 selected features ($\sigma$ =10.67619)* ........ 5-8

**Figure 5-5** *Error estimation of common $\sigma$ for a classification between upper and mixed case word images using 15 selected features ($\sigma$ =10.00000)* ........ 5-8

**Figure 5-6** *Error estimation of common $\sigma$ for a classification between lower and mixed case word images using 23 selected features ($\sigma$ =18.32981)* ........ 5-9

**Figure 5-7** *Comparison between the selected and 36 extracted features for seen data (common $\sigma$ )...* ........ 5-11

**Figure 5-8** *Comparison between the selected and 36 extracted features for unseen data (common $\sigma$)* ........ 5-11

**Figure 5-9** *Error estimation of common $\sigma$ for classification lower, upper and mixed case word images using 36 extracted features ($\sigma$ =38.97138)* ........ 5-13

**Figure 5-10** *Error estimation of common $\sigma$ for triple classification using 25 selected features ($\sigma$ =22.30060)* ........ 5-14

**Figure 5-11** *Comparison between the selected and 36 extracted features for seen data (common $\sigma$)* ........ 5-16

**Figure 5-12** *Comparison between the selected and 36 extracted features for unseen data (common $\sigma$)* ........ 5-16

**Figure 5-13** *Comparison between the selected and 36 extracted features for seen data (different $\sigma_i$)* ........ 5-22

**Figure 5-14** *Comparison between the selected and 36 extracted features for unseen data (different $\sigma_i$)* ........ 5-22

**Figure 5-15** *Comparison between the selected and 36 extracted features for seen data (different $\sigma_i$)* ........ 5-27

**Figure 5-16** *Comparison between the selected and 36 extracted features for unseen data (different $\sigma_i$)* ........ 5-28

## *Chapter 6*

**Figure 6-1** *All correct words regardless of rank* ........ 6-3

**Figure 6-2** *Error estimation of common $\sigma$ for a classification between legible and illegible handwriting using 36 extracted features ($\sigma$ =5.47436)* ........ 6-5

**Figure 6-3** *Error estimation of common $\sigma$ for a classification between middle and illegible handwriting using 36 extracted features ($\sigma$ =0.01386)* ........ 6-5

**Figure 6-4** *Error estimation of common $\sigma$ for a classification of middle and legible handwriting using 36 extracted features ($\sigma$ =7.11064)* ........ 6-6

**Figure 6-5** *Error estimation of common $\sigma$ for a classification between illegible and legible handwriting using 16 selected features ($\sigma$ =0.00066)* ........ 6-8

**Figure 6-6** *Error estimation of common $\sigma$ for a classification between middle and legible handwriting using 15 selected features ($\sigma$ =0.0015)* ........ 6-8

**Figure 6-7** *Error estimation of common $\sigma$ for a classification between middle and illegible handwriting word images using 13 selected features ($\sigma$ =0.0001)* ........ 6-9

*Figure 6-8* *Comparison between the selected and 36 extracted features using common $\sigma$ for seen data* ........ 6-11

*Figure 6-9* *Comparison between the selected and 36 extracted features using common $\sigma$ for unseen data* ........ 6-11

*Figure 6-10* *Error estimation of common $\sigma$ for a classification between legible, illegible and middle handwriting using 36 extracted features ($\sigma$ =0.001)* ........ 6-13

*Figure 6-11* *Error estimation of common $\sigma$ for a classification between legible, illegible and middle handwriting using 20 features ($\sigma$ = 22.22464)* ........ 6-14

*Figure 6-12* *Comparison between the selected and 36 extracted features using common $\sigma$ for seen data* ........ 6-16

*Figure 6-13* *Comparison between the selected and 36 extracted features using common $\sigma$ for unseen data* ........ 6-17

*Figure 6-14* *Comparison between selected features and 36 extracted features using different $\sigma_i$ for seen data* ........ 6-23

*Figure 6-15* *Comparison between selected features and 36 extracted features using different $\sigma_i$ for unseen data* ........ 6-23

*Figure 6-16* *Comparison between using the selected and 36 extracted features for style classification with unseen data* ........ 6-29

*Figure 6-17* *Comparison between using the selected and 36 extracted features for style classification with seen data* ........ 6-29

## *Appendix A*

*Figure A-1* *Some samples from the database, writer 1 (lower case)* ........ A-1

*Figure A-2* *Some samples from the database, writer 2 (lower case)* ........ A-1

*Figure A-3* *Some samples from the database, writer 3 (lower case)* ........ A-2

*Figure A-4* *Some samples from the database, writer 4 (lower case)* ........ A-2

*Figure A-5* *Some samples from the database, writer 5 (lower case)* ........ A-3

*Figure A-6* *Some samples from the database, writer 6 (lower case)* ........ A-3

*Figure A-7* *Some samples from the database, writer 7 (lower case)* ........ A-4

*Figure A-8* *Some samples from the database, writer 8 (lower case)* ........ A-4

*Figure A-9* *Some samples from the database, writer 9 (lower case)* ........ A-5

*Figure A-10* *Some samples from the database, writer 1 (mixed cased)* ........ A-5

***Figure A-11*** *Some samples from the database, writer 2 (mixed cased)* ........................................ *A-6*

***Figure A-12*** *Some samples from the database, writer 3 (mixed cased)* ........................................ *A-6*

***Figure A-13*** *Some samples from the database, writer 4 (mixed cased)* ........................................ *A-7*

***Figure A-14*** *Some samples from the database, writer 5 (mixed cased)* ........................................ *A-7*

***Figure A-15*** *Some samples from the database, writer 6 (mixed cased)* ........................................ *A-8*

***Figure A-16*** *Some samples from the database, writer 7 (mixed cased)* ........................................ *A-8*

***Figure A-17*** *Some samples from the database, writer 8 (mixed cased)* ........................................ *A-9*

***Figure A-18*** *Some samples from the database, writer 9 (mixed cased)* ........................................ *A-9*

***Figure A-19*** *Some samples from the database, writer 1 (upper case)* ........................................ *A-10*

***Figure A-20*** *Some samples from the database, writer 2 (upper case)* ........................................ *A-10*

***Figure A-21*** *Some samples from the database, writer 3 (upper case)* ........................................ *A-11*

***Figure A-22*** *Some samples from the database, writer 4 (upper case)* ........................................ *A-11*

***Figure A-23*** *Some samples from the database, writer 5 (upper case)* ........................................ *A-12*

***Figure A-24*** *Some samples from the database, writer 6 (upper case)* ........................................ *A-12*

***Figure A-25*** *Some samples from the database, writer 7 (upper case)* ........................................ *A-13*

***Figure A-26*** *Some samples from the database, writer 8 (upper case)* ........................................ *A-13*

***Figure A-27*** *Some samples from the database, writer 9 (upper case)* ........................................ *A-14*

## *Appendix B*

***Figure B-1*** *Percentage of lower case words, which are correctly or incorrectly classified (lower/upper data set) using all 36 feature* ........................................ *B-1*

***Figure B-2*** *Percentage of mixed words, which are correctly or incorrectly classified (upper/mixed data set) using all 36 features* ........................................ *B-2*

***Figure B-3*** *Percentage of lower words, which are correctly or incorrectly classified (lower/mixed data set) using all 36 features* ........................................ *B-2*

***Figure B-4*** *Percentage of mixed words, which are correctly or incorrectly classified (lower/mixed data set) using all 36 features* ........................................ *B-3*

***Figure B-5*** *Percentage of mixed words, which are correctly or incorrectly classified (upper/mixed data set) using 15 selected features* ........................................ *B-4*

***Figure B-6*** *Percentage of mixed words, which are correctly or incorrectly classified (lower/mixed data set) using 23 selected features* ........................................ *B-5*

***Figure B-7*** *Percentage of lower words, which are correctly or incorrectly classified (lower/mixed data set) using 23 selected features* ........................................ *B-5*

**Figure B-8** *Percentage of lower case words, which are correctly or incorrectly classified (upper/ lower/ mixed data set) using 36 features* ........ B-6

**Figure B-9** *Percentage of mixed case words, which are correctly or incorrectly classified (upper/ lower/ mixed data set) using 36 features* ........ B-7

**Figure B-10** *Percentage of lower case words, which are correctly or incorrectly classified (upper/ lower/ mixed data set) using 25 selected features* ........ B-9

**Figure B-11** *Percentage of mixed case words, which are correctly or incorrectly classified (upper/ lower/ mixed data set) using 25 selected features* ........ B-9

**Figure B-12** *Percentage of lower case words, which are correctly or incorrectly classified (lower/ upper data set) using 36 features* ........ B-10

**Figure B-13** *Percentage of mixed case words, which are correctly or incorrectly classified (mixed/ upper data set) using 36 features* ........ B-11

**Figure B-14** *Percentage of mixed words, which are correctly or incorrectly classified or misclassified (lower/ mixed data set) using 36 features* ........ B-11

**Figure B-15** *Percentage of lower words, which are correctly or incorrectly classified (lower/ mixed data set) using 36 features* ........ B-12

**Figure B-16:** *Percentage of mixed case words, which are correctly or incorrectly classified (mixed/ upper data set) using selected features* ........ B13

**Figure B-17** *Percentage of mixed case words, which are correctly or incorrectly classified (lower/ mixed data set) using selected features* ........ B-13

**Figure B-18** *Percentage of lower case words, which are correctly or incorrectly classified (lower/ mixed data set) using selected features* ........ B-14

**Figure B-19** *Percentage of lower words, which are correctly or incorrectly classified (lower/ upper/ mixed data set) using 36 features* ........ B-15

**Figure B-20** *Percentage of mixed words, which are correctly or incorrectly classified (lower / upper/ mixed data set) using 36 features* ........ B-15

**Figure B-21** *Percentage of mixed words, which are correctly or incorrectly classified (lower/ upper/ mixed data set) using 25 selected features* ........ B-16

**Figure B-22** *Percentage of lower words, which are correctly or incorrectly classified (lower/ upper/ mixed data set) using 25 selected features* ........ B-17

# 1. INTRODUCTION

Cursive script and machine printed character recognition have been the subject of intensive research for many years. As the amount of documents to be processed increases day by day, so the need for an automated way of extracting information from the documents or texts becomes ever greater. Unfortunately, the large variation seen in handwriting style makes the task of cursive script recognition very difficult [MADHVANATH01] [CONNELL02] [FAVATA01].

## 1.1 Machine Printed Writing Style Characterisation

Nowadays very high optical character recognition (OCR) rates are possible; particularly for text machine printed in clear, easy to read fonts [SRIHARI01] [MORI91][JUNG99]. However, even here there are problems. Garain and Haudhuri [GARAIN99] show that the deitalicization of italic words can produce a significant improvement on the recognition accuracy of a text recognition system. Baird and Nagy [BAIRD94] have also demonstrated that a significant improvement in the recognition accuracy of an OCR system could be achieved by utilizing the font information. Font recognition is thus a fundamental issue for automatic document processing. Font recognition can reduce the number of alternative shapes for each class leading to essentially single-font character recognition [NAGY00]. However, considering the number of fonts available, it is quite a difficult task and much effort has been spent towards achieving complete omnifont recognition [ZHU99][JUNG99]. All of this information thus points to the fact that the style of writing is very important for the recognition of printed writing.

## 1.2 Cursive Script Style Characterisation

The difficulties of style characterisation are even worse when handwriting is to be handled [IMPEDOVO91][POWALKA96][SCHOMAKER94]. Since optical character recognition methodologies for machine printed and handwritten texts are different, it is necessary to separate these two types before feeding to the recognition system in order to achieve optimal performance. Unfortunately, few papers exist in the literature on classification between machine-printed and hand-written text but from [KUHNKE95][PAL01] it can be understood that machine printed and hand-written character recognition differ quite substantially from each other. In addition although research into recognising hand-written characters and numerals has reached a development stage, the recognition of unconstrained cursive handwriting has proven to be much more difficult. The problem of handwriting recognition is far from being solved due to the vast variability in human handwriting both between different writers (inter-writers) and within the same writer (intra-writer) [LEEDHAM94] [MADHVANATH01] [CAMASTRA01].

To give an indication of how handwriting can vary between writers, figure 1-1 shows a selection of ways different writers can write the same word. Consequently, the recognition algorithm must deal with a variety of author specific writing.

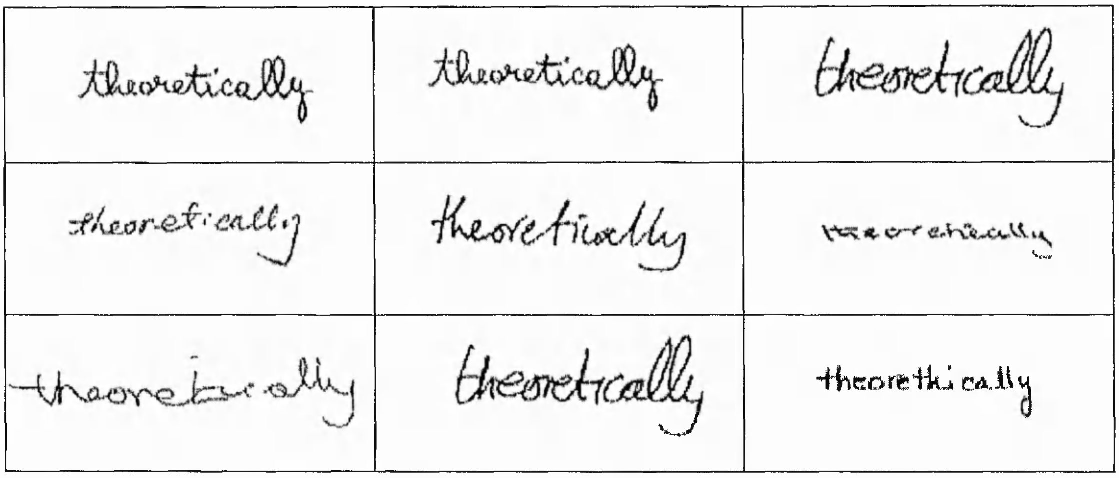


**Figure 1-1:** Samples of different writing styles.

Previous research has shown that writing style can vary significantly with geographical location, cultural background, age, sex and so forth [POWALKA95][CHA01]. Indeed people often completely redefine their style of writing as they age. The result is an enormous variability of handwriting. The characteristic of cursive handwriting such as height of ascenders or descenders, word length, letter concavities etc. make the different style of writing. In cursive handwriting, letters can be connected in a variety of ways and the letter standards can differ greatly; sometimes to the point where they can be totally illegible.

The two main approaches used in handwriting recognition have been identified as (1) global (whole word) approaches and (2) segmentation based approaches. For both global and segmentation based approaches, the diverse styles and size of handwriting play a large factor in the failure of current techniques. The extent of this variability is such that generalised algorithms based on detecting a set of common invariant features can only go so far in addressing the problem related to the recognition of off-line cursive handwriting. Furthermore the difficulties associated with variability have forced the imposition of artificial constrains such as disallowing the mixing of lower, upper, mixed case handwriting.

Cursive handwriting variability is not only due to writer's style but also to geometric factors determined by the writing conditions such as thickness of writing, which depends on the sharpness of the pen, the pressure exerted on it by writer etc. Experience with analysis of word recognition systems shows that it is unlikely that a system based on a single pattern recognition approach will be capable of handling the large variation and variability in human handwriting. The correction of this variability, prior to recognition, can be helpful in reducing the variability and can lead to an important improvement in recognition performance. Hence, for current handwriting recognition systems, a pre-processing stage is normally included. The aim is to remove unwanted variation and present, to the recogniser, characters that are as close as possible to the model templates. The main functions of such pre-processing steps are usually the correction of slant [DING99], the deskewing of hand-written words [BOZINOVIC89], normalisation [NICCHIOTTI97] etc. The use of these pre-processing steps has been shown to improve the image quality and correct the character string recognition. However, as part of this process, the original information may be lost.

Many attempts have also been made to deal directly with poorly written handwriting [HAMANAKA00]. Unfortunately, these improvements tend to result in a decrease in the performance of the recogniser system's ability to recognise clearly written characters. Currently, ambiguity of handwriting is considered by taking the context into consideration by using natural language processing to select words from the recognition list to improve recognition performance. For instance post processing ways of helping cursive script recognition overcome variation problem are used. These approaches do give limited success for improving the recognition performance but do eventually fail when the handwriting becomes highly illegible as far as the recogniser is concerned.

An alternative approach would be to select a recogniser that is suited to a particular style of writing. In this way special cursive-script recognition techniques would only be used when necessary and the original data would not be destroyed. Recognisers can be optimised for a small number of styles of handwriting. If a recognition system is designed to work for virtually any writer (a writer-independent recogniser) this large interclass variance will make the pattern class discrimination difficult. For this reason, some recognition systems have attempted to identify the different writing styles present in the data and model them separately. The Apple Newton was an example of such a system. Louis Vuurpijl and Lambert Schomaker [VUURPIJ96] present a technique for the automatic detection of generic writing styles such as "cursive", "handprint" and "mixed" (between cursive and handprint) etc. Such a system can be used to assign specialised recognition systems to a writer with an unknown writing style.

## 1.3 The Objectives of The Project

Several principal factors are concerned in designing and evaluating pattern classifier. Improving one of these characteristics such as, accuracy of classification process, the processing speed, robustness, memory space requirements etc, make the system particularly valuable. Providing adequate storage is usually a challenge in the design of image processing systems. Digital storage for image processing applications falls into three principal categories: (1) short term storage for use during processing (2) on-line storage for relatively fast recall and (3) archival storage, characterized by infrequent access. Recent efforts have concentrated on reducing the system complexity and computation cost and increasing the system efficiency (speed and extraction rate).

In the work presented here, it is concentrated on accuracy or efficiency of a system. It is hypothesised that one way of helping a cursive script recognition system would be to detect cursive writing style prior to the recognition stage. In this way the best recogniser could be selected for the style of writing using a prediction of legibility based on a given recogniser's performance. For this purpose, style classification has been broken down into case and quality of handwriting. Case classification of handwriting could be used as a means of limiting the size of the template database for word recognition. The quality of handwriting has been defined in term of its legibility. It is proposed that this approach would lead to determining the legibility of an unknown sample prior to recognition. In this way it would be possible to select the most suitable recogniser for the given handwriting sample.

The aims of this research are:

(1) To address the problem presented by lower, upper and mixed case variation in unconstrained cursive handwriting. It is proposed that a pre-classification of handwriting

could be employed to reduce the recogniser's search space (lexicon) in order to improve the overall recognition rate.

(2) To focus on the problem of classifying word images as legible, illegible or middle (between legible and illegible) prior to the recognition stage. In this way the best recogniser could be selected for each style of writing using a prediction of legibility based on the given recogniser's performance.

As there is no evidence on literature for case classification of cursive script handwriting word recognition and legibility of handwriting a novel approach of Multiple Discriminant Analysis (MDA) and Probabilistic Neural Network (PNN) will be used for this purpose. In this thesis the use of MDA for case and legibility classification is firstly considered. A PNN based on Bayesian decision and a Parzen estimator for estimating the density function is then used for the same purpose. This allows for a comparison between the two classification techniques to be given. The expected key advantage of the PNN techniques over the MDA technique are (1) the decision surface is guaranteed to approach the Bayes-optimal decision boundaries as the number of training samples grow, (2) The shape of the decision surface can be made as complex as necessary, or as simple as desired by choosing the appropriate value of smoothing parameter, (3) erroneous samples can be tolerated. Our proposed pre-classifier method produced instantaneous result and the only time consuming component is the in avoidable training part. The PNN, however usually trains orders of magnitude faster than multiple layer feed forward networks (MLFNs). In this case the computational expenses will be reduced.

The following publications describe the novel approaches pertaining to handwriting recognition that have been devised within this work:

[EBADIAN99] M. Ebadian Dehkordi, N. Sherkat and R.J. Whitrow, "Classification of Off-line Handwriting Words into Upper and Lower Case", Document Image Processing and Multimedia, IEE, London, March, 1999.

[EBADIAN99] M. Ebadian Dehkordi, N. Sherkat and R. J. Whitrow, "A Principal Component Approach to Classification of Handwriting Words", Fifth International Conference on Document Analysis and Recognition (ICDAR'99), 781-784, India, September 1999.

[EBADIAN00] M. Ebadian Dehkordi, N. Sherkat and Tony Allen, "Case Classification of Off-line Handwritten Words Prior to Recognition", Fourth International Conference on Document Analysis and System (DAS'00), 325-334, Rio de Janeiro, December 2000.

[EBADIAN01] M. Ebadian Dehkordi, N. Sherkat and Tony Allen, "Prediction of Handwriting Legibility", Sixth International Conference on Document Analysis and Recognition (ICDAR'01), Seattle, September 2001.

[EBADIAN02] M. Ebadian Dehkordi, N. Sherkat and Tony Allen, "Prediction of Handwriting Legibility", Accepted to be published in international journal on document analysis and recognition (IJDAR).

## 1.4 Outline Of The Thesis

The thesis is structured into seven chapters.

Chapter 2: Provides a review of the state of the art in writer style classification. The problems of handwriting recognition connected to the style characterisation are also discussed.

Chapter 3: Presents the 36 handwriting features that are to be extracted from handwriting word images for the purpose of case and legibility classification.

Chapter 4: Outlines the multiple-discriminant analysis (MDA) based and probabilistic neural network (PNN) based classification techniques. In the case of the PNN system a parzen estimator is used to estimate the density function of each class and a leave-one out method is used for training. Finally the efficiency of each feature is also calculated by using the MDA technique to extract the best features for both case and legibility classification.

Chapter 5: Presents the experimental results and analysis of case classification using the MDA methods for binary classification (upper/lower, upper/mixed, lower/mixed). The experimental result and analysis using the PNN method with common $\sigma$ and different $\sigma_i$ values for both binary and triple classification (upper/lower/mixed) are then given. Finally a comparison between all classifiers (MDA, PNN with common $\sigma$ and different $\sigma_i$ values) is made.

Chapter 6: Presents the experimental results and analysis of handwriting legibility classification using the MDA methods for binary classification (legible/illegible, legible/middle, illegible/middle). The experimental results and analysis of using the PNN method with common $\sigma$ and different $\sigma_i$ using binary and triple classification (legible/illegible/middle) is also given. Finally a comparison between all classifiers (MDA, PNN with common $\sigma$ and different $\sigma_i$ values) is made.

Chapter 7: Presents the conclusion and future work.

# 2. LITERATURE REVIEW

This chapter provides an overview of the problems associated with the automatic classification of style of writing; specifically in off-line cursive script handwritten images. In addition, current techniques and methods proposed for dealing with this variability in cursive script recognition systems are critically reviewed.

In order to help define the scope of the problems associated with style classification, section 2.1 reviews the techniques and problems involved in style classification for optical character recognition systems. Section 2.2 then introduces the properties of handwriting. This is followed by an overview of handwriting recognition techniques and their problems in section 2.3. Section 2.4 describes the different style classifications techniques and recognition improvements that have been reported in the literature. Finally, the ideas proposed in this work are introduced and related to the unresolved problem of robust style classification for cursive script recognition.

## 2.1 Optical Character Recognition

Optical character recognition (OCR) is a character-based recognition technique that is capable of recognizing machine printed fonts and alphanumeric handprints. The recognition is based on matrix or template matching technique where each character is compared to a set of prototype characters in the database. However, shape discrimination between characters that look alike is difficult for machine recognition. Some characters have similar shapes, such as U-V, C-L, a-d, n-h. Similar shapes also occur between certain characters and numbers, O - 0, I - 1, l - 1, Z - 2, S - 5, G – 6, etc. Some of these pairs, such as I – 1, can be written identically. They can only be distinguished by context. Also, many upper and lower case characters have similar shapes: C-c, K-k, O-o, etc. For most of these pairs, the distinguishing factor is the character size relative to the line spacing or to other character sizes. For others, such as P-p and Y-y, the distinction depends primarily on the position of the character relative to the baseline. In addition to this inherent character ambiguity, recognition accuracy often drops significantly when a document contains different fonts. The reported recognition rate varies widely according to the quality of the input document, the fonts used in the document, the presence of proportional spacing and so on. However recognition rates of omnifont commercial OCR systems usually stay at around 99% [MORI91] [IMPEDOVO91].

The following aspect of printed text has been exploited to facilitate recognition [HO01] and shows how style of machine printed writing is important in performance of OCR.

1. Consistency of the shape of individual characters within a document.
2. Font-independent characteristics, such as ascenders, descenders, relative size and vertical position.
3. Ease of character and word-level segmentation in many documents.
4. Stability of symbol-n-gram frequencies across documents.

5. Prevalence of lexicon words, which constitute a small fraction of all possible combinations of symbols.
6. Partial recognition of the text by an omnifont recognition engine.

Research into the recognition of printed written characters or words has reached a mature stage and algorithms for identification of different styles of machine printed writing (such as italic, bold, capitalised etc) and high accuracy are reported in [CHAUDHURI98][PLAMONDON00]. Detecting these type styles helps in the automatic extraction of lines containing title, authors' name, subtitles, captions, table title and references as well as identifying sentences that have important terms occurring in the text. The italic, bold and capitalised type written words can be identified by measuring the slant angle and relative stroke thickness as well as using zoning information [GARAIN99][IMPEDOVO91]. In this work, the slant angle of the word is first computed then, based on the slant angle, the words are de-italicised by an inverse operation. This research shows that the deitalisation of italic words produces a significant improvement in recognition accuracy. By adapting this technique the overall miss-recognition rate given by an existing OCR system for italic words, has been reduced from 6.85% to 0.33%[GARAIN99][IMPEDOVO91]. This information could also increase the accuracy in extracting figure, captions and table titles.

Another attempt has been made by Ho [HO01]. She presents strategies and results for identifying the symbol type (lower case, upper case, digit and punctuation or special symbols) of every character in a text document by using various kinds of information from neighbouring characters. Eleven numerical features describe each character cluster. Each feature is a single frequency estimated from a whole document such as bigram diagonal, trigram diagonal, length word etc. details of these features are shown in [HO01].

These vectors of eleven elements were standardized and then used in a nearest neighbour classifier using Euclidean distance. In this paper reliable segmentation and shape clustering is assumed to determine the benefits of contextual information under ideal conditions. Two classifiers are examined. Classifier 1 used the training set as if it were a very long document, i.e., all features were calculated using one single set of bigrams and trigrams, and there are 78 reference (training) vectors corresponding to 78 observed symbols. Classifier 2 used the training set as separate articles, i.e., bigram and trigram frequencies were calculated on a per-article basis, so there were 18604 reference (training) vectors (many symbols did not occur in each article). Both classifiers were tested on each test article as well as the entire test set treated as one long article. With the test set as one single, long article of 298K words, the overall correct rate using classifier 1 is 99.96%, (298042/298160). The rate for classifier 2 is 93.34% (278305/ 298160). Thus classifier 1 is seen to be more accurate, although both classifiers are far better than the default assignment of every cluster to the type LOWER (accounting for 88.72% of all characters). Classifier 1 made only 2 errors (assigning 'X' to LOWER and '(' to UPPER). Classifier 2 made more diverse errors and the most common type was the assignment of uppercases {B, C, D, E, H, M, O, and W} to LOWER.

The detection of the font and style can improve the character segmentation as well as the character recognition because the identification of the font provides information on the structure and the typographical design of characters [JUNG01][KAHAN87]. The above information shows how style of printed writing can affect the recognition result. The effect is even greater when cursive handwritten characters or words are to be handled.

## 2.2 Properties of Handwriting

A written language has *n* alphabet of characters (or letters), punctuation symbols etc. Handwriting consists of a time sequence of strokes where a stroke is the writing from pen down to pen up. The characters of writing are usually formed in sequence, one character being completed before the next is started, and the characters typically follow some spatial order, i.e left to right. The position and size of the letters are important. Uppercase letters sit on the baseline and are full size. Lowercase letters are smaller and are about half the height of upper case letters. Some lower case letters have an ascender, which extends upward to almost the height of the uppercase letters. Some have a descender, which extends down below the baseline and some have both.

All characters vary in both their static and dynamic properties. Static variations can occur, for example, in size or shape. Dynamic variations can occur in stroke number and order. The degree of variation depends on the style and speed of writing, with hasty writing usually showing the greater variation. This variety makes the task of handwriting recognition very difficult. Consequently, the recognition algorithm must deal with a variety of author-specific idiosyncrasies. Moreover, there is little or no control in most off-line scenarios on the type and instrument used. The artefacts of the complex interactions between instrument and subsequent operations such as scanning and binerization present additional challenges to the algorithms used for off-line handwriting recognition. In particular, low-quality images, where poor image quality such as broken lines, are produced by the machine printers or fax machines, pose a serious challenge to current pattern recognition techniques.

Although some research has been done on broken handwriting [WANG99][HU99] the illegibility of poor writing or broken characters always creates a problem in handwriting recognition. Coates in 2001 [COATES01] [TURING50] proposes a method in order to overcome the gap between human and machine vision system. The choices of image degradation are thickened images, thinned images, noisy images, condensed fonts and italic fonts. The range of values for two of degradation parameters are blurring and thresholding. This paper shows how the choice of these ranges can ensure that the images are legible to human reader but illegible to several of the best present day optical character recognition machines. Each OCR machine in this experiment was sensitive to slight changes in the parameters. For example, one machine's accuracy dropped from 40-50% to 0% when the threshold fell from 0.04 to 0.02 (blurring=0.8). It also dropped from 28% to 0% when blurring fell from 0.4 to 0.0 (threshold=0.04); this change being barely perceptible to the human eye. Unfortunately, such approaches will not be able to say which images thought legible to human readers are illegible to several of the best present day optical character recognition systems. Therefore legibility should be defined in terms of specific recogniser.

## 2.3 Handwriting Recognition Techniques

Off-line cursive script recognition (CSR) remains an extremely challenging task due to the vast variety in handwriting. There are three different approaches for CSR.

1. Analytical or segmentation based approach
2. Word-based or holistic approach
3. Combining the results of above approach

Analytical or segmentation approach treats a word as a collection of simpler subunits such as characters and proceeds by segmenting the word into these units [VINCIARELI00][CASEY95]. Thus, this approach has to deal with the problems of segmentation ambiguity and variability of segment shape.

Berrin in 1994 and 1998 [BERRIN94][BERRIN98] showed that the difficulty in segmentation could be credited to the style of writing. One such ambiguity is due to the ligature shaped strokes that appear in most letters. A ligature can connect two letters in any of the following ways. Letters can be connected to the following letter from the bottom of their body, from the top of their body or from their descender. The shape or location of a ligature is therefore not unique. This variety of styles makes it more likely that ambiguity can occur in the segmentation of cursive script. Thus, words are always either over or under segmented. Style of writing is therefore important in the segmentation process. [BERRIN94] [BERRIN97] also show that it is important to have not only dominant slant but also letter width, pen thickness for segmentation. In his work he uses straight lines in eight fixed angles (-30, -20, …, 40 wise clock from the vertical) to divide letters and a term letter boundary for using the ligature between two letters.

Holistic approaches extract holistic features from the word image and use the features directly to arrive at the word identity. In order for this feature-level matching to be possible, every candidate from the lexicon must have a feature representation similar to that used to represent the image features. These features can be characterised into three classes: (1) High level features such as ascenders, descenders, loops, word length, dots, holes etc [WAARD95][LEROUX91], (2) Intermediate level features such as edges, end-points, concavities, diagonal and horizontal strokes [BROWN80] and (3) Low level features such as stroke direction distribution [YAMAMOTO84][HULL91]. At the moment there is no single feature vector that can be considered as optimal. Holistic approaches circumvent the issues of segmentation ambiguity and character shape variability that are primary concerns for analytical approaches and may succeed on poorly written words where analytical methods fail to identify character content. Holistic approaches have been used traditionally in applications wherein the classes are few and fixed. For example, the cheque amount recognition task.

Analytical and holistic methods can complement each other's strengths and provide for a robust system [DODEL95][VINCIARELLI00] [HUANG93][POWALKA96]. This method is more robust and independent of segmentation issues since the recognition of all letters in the word is not necessary. However the system is based on distinguishing between words in a lexicon.

In summary, progress in off-line recognition of isolated characters achieved in the past years is quite remarkable. In two surveys, recognition rate of up to 99.5% and reliability of up to 100% have been reported [SUEN92][SUEN93]. By contrast, the problem of off-line CSR is still widely unsolved. The recognition rates reported in the literature vary between 50% and 96% depending on the experimental conditions and task definition. Recognition rate of 98% on the word level has been achieved in experiments with cooperative writers using two dictionaries of 150 words by [BUNKE94] and a high level of performance is observed by combining the results of both above approaches (analytical and holistic) [HUANG93][POWALKA96]. Unfortunately, it is almost impossible to compare the recognition performance achieved by the different systems for the following reasons. Firstly, many methods use proprietary databases or are tested on relatively small lexicons. Secondly, the recognition performance of a system relies on many factors such as pre-processing, post-processing and segmentation. The chosen lexicons and recognition methods used also inevitably affect the final recognition performance. Papers that attempt to comprehensively compare the recognition result can be found in [VERMA98][SIMNER96].

Although much research has been done in order to address reliable CSR, current results are far from satisfactory. The greatest difficulty in CSR is due to the fact that cursive writing has great variance in style. This variability is generally explained by personal, emotional, international circumstantial factors. Srihari, Cha, Arora and Lee [SRIHARI01] undertook a study to objectively validate the hypothesis that the handwriting is individualistic. Validation of individuality was done using two different approaches, both based on classificatory models: (1) Identify the writer from a set of possible writers (2) verifying whether two documents were written by the same writer. Their paper shows that writer identification accuracy can be achieved to 98% for two writers. In the verification approach, the features were mapped onto the feature distance domain, and the individuality problem was tackled as a 2-class classification problem within and between author distances. Verification accuracy was about 95%. Section 2.4 summarizes some of the strategies and methods that have been published in an attempt to help improve the recognition performance by introducing different styles of writing.

## 2.4 Style Classifications and Recognition Improvement

Tapper [TAPPER84] described the range of pattern recognition problems by the severity of letter segmentation. He introduced several significantly different types of handwriting such as;

- Boxed discrete characters;
- Spaced discrete character;
- Run-on discretely written characters;
- Pure cursive script writing;
- Mixed cursive script writing.

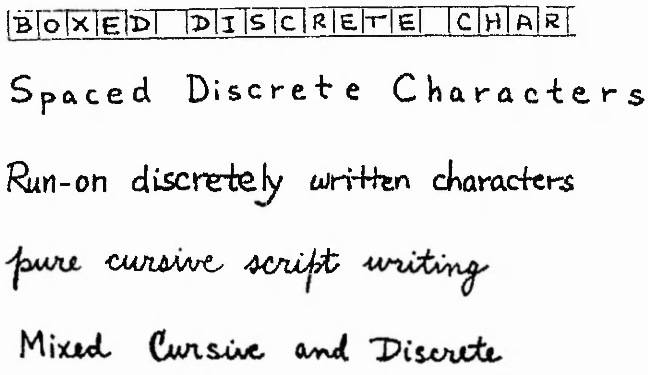



**Figure 2-1.** Types of handwriting as defined by Tapper [TAPPER84].

Figure 2-1 shows types of handwriting as defined by Tapper [TAPPER84]. This research shows that discrete characters written in boxes require no letter segmentation. Spaced discrete characters require spatial letter segmentation and pure cursive script require a lot of segmentation. Generally, the more the characters are touching with each other the lower will be the recognition rate. This paper shows that the recognition results on spaced discrete characters are promising. The low segmentation error indicates that the spaced discrete problem is only slightly more difficult than the boxed discrete one. The recognition text consisted of 72 words containing 325 characters with good distribution among the upper and lower case letters and digits. Recognition was performed using the prototypes obtained from the same writer. Recognition accuracy for a spaced-discrete recogniser with a full and selected (upper, lower and digit) alphabet is 94.1% and 98.3% respectively. In this research, the experiment on cursive writing, results were obtained for carefully produced writing samples from three writers. For each writer a set of 165 letter prototypes was established by adding 113 letters extracted from the cursive writing of a specific text to 52 prototypes from discretely written cursive characters. The recogniser was tested on new samples of cursive written for each writer, the accuracy of recognition is 96.6%. The recognition results on cursive writing, though based on small samples of writing using lower case words only, indicate that elastic matching is also a promising technique for recognition of cursive writing.

In 1993, Powalka [POWALKA93] suggested the use of handwriting style features to isolate some of the specific characteristics of a writing sample in order to minimize the ambiguity. In this approach, the style information was to be extracted and used prior to actual recognition. Again in 1995, Powalka [POWALKA95] used the characteristics of handwriting to guide the combination process of recognition results obtained from a multiple approach. In this approach the style information is used after the recognition. However, in both approaches handwriting style is to be analysed inherently by the recognition system without taking into account the knowledge of the hand-written context.

Some handwriting words are usually slanted or italicised due to the mechanism of handwriting. One way to tackle this problem is to minimise the variability by introducing two schemes: (i) standardisation of raw data by normalisation and slant correction (ii) constraining the problem by restricting the number of objects to be recognised. Bozinovic and Srihari [BOZINOVIC89] and Kim Govindavaju [KIM97] have proposed slant correction techniques where the average slant is estimated from the angles of extracted vertical strokes. Guillevic and Suen [GUILLEVIC94], Kavallierataou [KAVALLIERATOU00], Nicchiotti and Scagliola [NICCHIOTTI97] analysed a set of projection histograms for the estimation of the average slant angle. Kimura [KIMURA93], Simoncini and Kovacs [SIMONCINI95], Ding [DING00] and Britto [BRITTO00] utilized statistics of chain-code stroke contours. These papers all make the assumption of constant slant throughout a word. Although, these methods give a good estimate of the word slant, the slant often tends to be underestimated or depends on the skew of the writing. A more widely acceptable assumption is that the slant angle fluctuates in a word due to various factors such as writer's habit, the inherent shape of each character, and writing position. This assumption raises the necessity to estimate local slant angle and to correct them non-uniformly. Some researchers have attempted to deal with such variability by using Hidden Markov Models [UCHIDA01][CHEN93]. Uchid in 2001 [UCHIDA01] shows the present technique provides near-perfect correction while the non-uniform slant correction techniques fail. The present technique sometimes over-corrects the slant of several alphabets. This over-correction is still an open problem of the present technique.

All the above efforts have been made to reduce the variability of writing. However since variability of handwriting is an inherent property of human beings, researchers are now looking in the direction of using a preliminary step for writing classification in order to take the variability into account rather than trying to overcome it [CHEN92][CRETTEZ95]. This means that each recognition system should adapt itself to a given handwriting style by pre-processing the handwriting in order to identifying the specific type or family of the handwriting style.

In an attempt to analyse the variability of handwriting, Crettez [CRETTEZ95] in 1995 described measures to characterise a writer's style. Thickness of writing, which depends on the sharpness of the pen and of the pressure exerted on it by writer, is one measure that was used in this research. The number of letters per unit length and the numbers of vertical strokes encountered in the middle zone are two other measures. For each word the normalised histogram of the different straight line parts of a tracing are drawn as a polar diagram (named directional diagram). These directional diagrams are then segmented into different directional lobes in such a way that it is possible to segment a histogram into different Gaussian modes. A set of lobes constitutes a good characteristic of handwriting. In his work a variability space is defined by applying these measures to the words of a database. Using fuzzy clustering, he regroups handwriting styles into a small number of specific families. The "unity of belongingness" can be defined as the fact that all the words of a given amount mainly present the same degree of membership either to the same family, or to the same sub-set of families. The words of the first family are oriented to the left. Their direct ligatures are degenerated into a horizontal segment. The second family is the upright handwriting with equilibrate ligatures and with a thick tracing. The words of third family have also an upright handwriting, but they present a higher spatial periodicity than those of the second family, and the reverse ligature is weak or absent. Whereas the fourth family is strongly oriented to the right with a lack of retroactive ligatures and a high spatial density. Such a handwriting pre-processing would facilitate some models of word for Hidden Markov Models (HMM) [SCHOMAKER94].

Another attempt to analyse the variability of handwriting was done by Gilloux [GILLOUX94]. [GILLOUX94] describes a method for improving hand-written word recognition by implicitly recognizing the style of the writer. This method is applied in the general framework of HMM. The proposed method makes use of a set of models rather than of a unique model for each word and the writing style is automatically detected during recognition. In his paper, writing styles are classified based on a distinction between word shapes: cursive script vs. hand-printed words, run-on vs discrete words, differences in skew angle values, stability of lower and upper extensions of letters and presence or absence of loops in naturally looped characters. A Hidden Markov Model (HMM) is then used to represent this problem as a stochastic model. Therefore, one of the consequences of this recognition is the implicit detection of the writing style. This is a relatively new and promising direction of research in the automatic recognition of cursive handwriting. Preliminary results of his work show that the implicit identification of writer style enhances performance in widely varying types of handwriting. In other words, the proposed method allows improving hand-written word recognition by detecting writing style at recognition time. The word recogniser used in this work was trained on a set of 7648 images of handwritten city names extracted from live mail addresses (3831 handprinted, 3817 cursive). The method has been tested on a different test set of 4090 words (2045 handprinted, 2045 cursive). In this test, a dynamic lexicon of 10 names was generated by adding 9 random names extracted from a list of 8469 different city names to the correct interpretation of each test pattern. The ratio of correct interpretation is measured in a set of $n$ candidates (n=1,2,3,5). The result is reported in table 2-1.

| | | 1 | 2 | 3 | 5 |
|---|---|---|---|---|---|
| Whole test set | 1 style | 84.4 | 90.5 | 93.2 | 95.8 |
| | 2 style | 87.0 | 91.7 | 94.4 | 95.8 |
| Cursive words | 1 style | 85.1 | 92.4 | 93.9 | 97.3 |
| | 2 style | 86.1 | 92.9 | 94.4 | 97.3 |
| Handprinted words | 1 style | 80.2 | 88.5 | 91.8 | 95.8 |
| | 2 style | 82.7 | 90.1 | 93.3 | 95.8 |

**Table 2-1.** Word recognition result on the test set. R(n): correct answers in a list of length n.[Gilloux94]

One of the other factors that has been used to classify writing style is the neatness of handwriting, where neat writing is defined as a handwritten word in which the word's slant, letter skew and instances of the same letter at different positions are relatively constant [LEEDHAM94]. [BOULETREAU97] presents a new family of parameters for handwriting analysis based on the fractal behaviour of writing. These parameters also allow the classification of handwriting into different families. These parameters qualify a particular aspect of the writing. In this paper a legibility graph then allows a formulation of legibility definition. N. Vincent and T. Freche [VINCENT01] also defined new parameters that allow the qualification of some handwriting properties. Two properties are presented, regularity of the line drawn and regularity of the pattern involved in the writing using fractal models. Their paper shows that these fractal parameters are suitable for the qualification of complex entities. The parameters correspond respectively to very regular writing and to irregular writing. However, it is difficult to compare the performances of recognition systems, as there are no quantitative measures of neatness or definition of size of writing etc and all of the researchers used different methods to classify the style of handwriting.

Sung-Hyuk Cha and Sargur N. Srihari [CHA01] present a datamining technique to mine a database consisting of experimental and observational unit variables. Experimental unit variables are those attributes which make sub-categories of the entity (e.g. demographic data) and observational unit variables are the features used to classify the entity e.g. handwriting styles etc. In order to build a machine that can classify an unseen instance into its sub-category, each class (subgroup) must have a substantial number of instances for the sake of valid statistical inference. This is called support. For this purpose a priori algorithm is applied to select only sub-categories that have enough support among all possible ones in a given database. An artificial Neural Network classifier is then used to discriminate between selected sub-categories. Finally, the performance measures for each selected sub-category problem are reported as the final output. This method was used to determine the similarity of handwriting style within a specific group of people such as male or female writer. For males in the age 15-24 group or white females in the age 45-64 group an 87% correct classification was observed.

## 2.5 Future Direction

All of the reported techniques have been shown to be able to improve the overall recognition performance. However, a direct comparison between them is impossible due to the different style definitions and testing environments. In addition, the reported results indicate that all of these techniques are incapable of completely solving the problem of variability in writing on their own. Thus, the reported work suggests that before it is possible to use a pre-processing style classification technique to improve recognition performance, a robust style classification technique needs to be developed. As each writer and each word has its own style of writing and as each recogniser has its own features to recognise words it can be hypothesised that some words could be better recognised by style specific recognisers rather than generic (all style) recognisers. Indeed, the results presented in section 2.3 show that specialized word recognisers are smaller, faster and can achieve similar or better recognition results than generic recognisers.

Coates [COATES01] has shown that there are a variety of images, which though legible to a human reader are illegible to several of the best optical character recognition systems. By defining the legibility of handwriting, based on the performance of a given recogniser, we show that it is possible to detect writing style prior to the recognition stage in order to choose the best recogniser for the given writing style. Our method does not change the word to suit the recogniser but aims to find the best recogniser to suit the words. In this research we also show that a pre-classification of words into upper, lower and mixed case could provide a useful means of reducing ambiguity. By successfully classifying the case of words prior to recognition the size of the lexicon used for any individual word recognition could be reduced which in turn should improve the recognition results.

## 2.6 Summary

This chapter provides an overview of existing style classification techniques, CSR methods and their applications presented in literature. In the first section optical character recognition systems and style of printed writings was discussed. The implementation techniques and their problems were briefly discussed. And it has shown that identification of the font provides information on the structure and the typographical design, which could improve segmentation and recognition steps.

The second section introduced the property of handwriting and shows how style of handwriting or variability of writing can be changed by human. This variability is due to the characteristic of the word and ligature shaped strokes, their position, etc. For instance the ligature shape in uppercase words differs with the same word in lower case words.

The third section presents general CSR approaches (analytical, holistic and a combination of these two approaches) and provides information of how variability could make these approaches difficult. The characteristic of the words, variability of handwriting and low CSR result motivated research to develop style classification prior to the recognition, such as case classification, which is presented in this research.

The fourth section has focused on style classification of handwriting and how style could improve the CSR performance. Most of the reviewed systems reduced the complexity of style of writing using pre-processing such as slant correction, normalisation, etc. Consequently, the original handwriting information is lost. This limits the potential of using pre-processing without losing original information. Some researchers [SCHOMAKER94][SCHOMAKER99][VUURPIJL96] have approached the problem of writer classification from a high level point of view. They aim to separate the writing style at coarser level. Their definition of style variability is limited to cursive, printed and mixed. The work presented in this thesis is another attempt in the direction of introducing a new and robust pre-recognition writing style classification in order to choose recognition methods better suited to the patterns themselves. However, in the case of CSR the reported results indicate that current cursive script style classification techniques fall short of being a complete solution to this problem. This work is an attempt to address this shortfall.

# 3. FEATURE EXTRACTION

## 3.1 Introduction

As a major factor influencing classification performance, features play a very important role in handwriting classification. This led to the development of a variety of features for handwriting classification [TRIER96][SRIKANTAN96]. In this section a number of features are introduced that can be used for both case and legibility classification. The extracted features tend to extract the different characteristics available in each word. The approach that is taken in this research is to firstly extract as many features as possible from each word. This is in order to represent the different characteristics of the word. The efficiency of those features in terms of their contribution to the style classification is then assessed based on a feature selection scheme introduced in the next chapter.

## 3.2 Contour-based Features

As a starting point, based on human perception of style, it was assumed that the word contour, as defined by tracing around the outside of the whole word, could contain information about the relationship of the underlying characters used in constructing the word [CHIEN98]. We extend this to the hypothesis that the 'synergy' within the word resulting from the way in which the neighbouring characters follow/influence each other is encapsulated in the word shape. A number of features were therefore introduced which are based on the contour of the handwritten word images. Using a single feature type has shown a certain limitation in achieving satisfactory classification performance and this leads us to use multiple types of feature.

A hand-written word can be described as a sequence of disjointed loop contours

$WI = \{C_i \mid C_i \cap C_j = \phi, i \neq j, j = 1,2,\ldots,N\}$.

Where N is the number of loop contours.

Each loop contour $C_i$ is a sequence of consecutive points on the x-y plane:

$$C_i = \{p_j \mid j = 1,2,\ldots,M_i, p_1 = p_{M_i}\}$$

where $p_1$ and $p_{M_i}$ are the end points of the $i^{th}$ loop contour.

The contour-based features used in our system are mainly based on:

(a) The chain coding from the eight primitive directions given by Freeman encoding [FREEMAN61].

Figure 3-1 refers to the eight primitive directions $d_i$ and represents the writing direction from a start point to an end point by following the upper outer contour of the word. Each loop contour $C_i$ can be represented by a chain code sequence

$$D_i = \{d_j \mid j = 1,2,\ldots,M_i - 1\},$$

and

$$D = \bigcup_{i=1}^{N} D_i$$

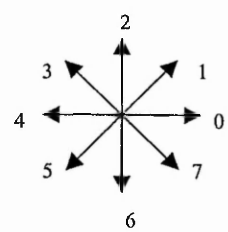



**Figure 3-1:** Eight primitive directions.

(b) Consecutive exterior angles and contour angles formed by pairs of vectors along the word images.

Figure 3-2 shows the exterior angle $a_l$ at point $p_l$ formed by a pair of vectors $d_l$ and $d_{l-1}$, and is located on the left-hand side of the vectors. The value of $a_l$ can be obtained easily using lookup Table 3-1. The sequences of exterior angles in a loop contour, $C_i$, is calculated as:

$$A_i = \{a_j \mid j = 2,3,\dots,M_i - 1\}$$

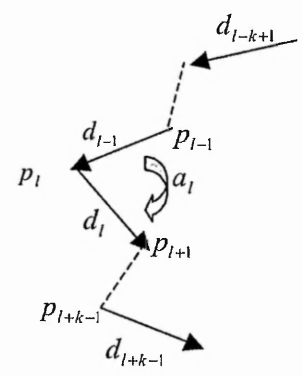



**Figure 3-2:** Angle $a_l$ at point $p_l$.

| $(d_{l-1} - d_l) \bmod 8$ | 0 | 1 | 2 | 3 | 5 | 6 | 7 |
|---|---|---|---|---|---|---|---|
| $a_l$ | 180 | 135 | 90 | 45 | 315 | 270 | 225 |

**Table 3-1:** $a_l$ as a function of $(d_{l-1} - d_l)$.

(c) Dominant points.

Dominant points refer to points of the following types:

(1) End points of the segmented regions of each individual loop contour.

(2) Points corresponding to local extreme of curvatures of each individual loop contour.

(3) Midpoints between two consecutive points of type (1) or (2).

Using concepts (a) to (c), the following subsections define the selected features in detail.

### 3.2.1 Global Features

[MADHANATH01] shows how word shape contains sufficient information to classify words in a certain lexicon. In the work it is also noted that if a word is written entirely in uppercase, there are no prominent or marked shape features present. Here it was also hypothesised that upper case words would have more straight lines in their contour than do lowercase words [EBADIAN99a][EBADIAN99b]. These characteristics of handwriting are different from one writer to another writer. A number of features based on the overall shape of a given word have been nominated. Assume $N$ is the number of loop contours.

*(1) An estimate of number of sharp angles in the whole word:* Ratio of number of original sharp angles to the total number of angles (**ROSP**):

$$\textbf{ROSP} = \frac{\sum_{i=1}^{N} card(A_i^{90})}{card(P)}$$

Where

$$A_i^{\theta} = \{a_j \in A_i \mid a_j \le \theta, j = 2 \ldots M_{i-1}\}$$

$$P = \bigcup_{i=1}^{N} C_i$$

$$card(P) = \sum_{i=1}^{N} card(C_i) = \sum_{i=1}^{N} M_i$$

and $card$ stands for the number of members in a set and sharp angles are the angle less than or equal to 90 degree.

*(2) An estimate of the component length (disjoint loop contours) or averaged component length* (**ACOL**):

$$\textbf{ACOL} = \frac{card(P)}{N}$$

*(3) Ratio of Vertical direction (2 and 6 directions given by Freeman code) to the total original chain code* (**RVO**):

$$\textbf{RVO} = \frac{card(N^{ver})}{card(P)}$$

Where

$$N^{ver} = \bigcup_{i=1}^{N} N_i^{ver}$$

$$N_i^{ver} = \{\mathrm{d}_j \in D_i \mid d_j = 2 \vee d_j = 6\}$$

and $card(N^{ver}) = \sum_{i=1}^{N} card(N_i^{ver})$ as $N_i^{ver} \cap N_j^{ver} = \phi \quad for\ i \neq j$.

*(4) Ratio of Horizontal directions (any 0 and 4 directions given by Freeman code) to the total original chain code* (**RHO**):

$$\textbf{RHO} = \frac{card(N^{hor})}{card(P)}$$

Where

$$N^{hor} = \bigcup_{i=1}^{N} N_i^{hor}$$

$$N_i^{hor} = \{d_j \in D_i \mid d_j = 0 \vee d_j = 4\}$$

and $card(N^{hor}) = \sum_{i=1}^{N} card(N_i^{hor})$ as $N_i^{hor} \cap N_j^{hor} = \phi \quad for\ i \neq j$.

*(5) Ratio of diagonal directions (any 1,3,5 and 7 directions given by Freeman code) to the total original chain code* **(RDO):**

$$\textbf{RDO} = \frac{card(N^{dia})}{card(P)}$$

Where

$$N^{dia} = \bigcup_{i=1}^{N} N_i^{dia}$$

$$N_i^{dia} = \{d_j \in D_i \mid d_j = 1 \vee d_j = 3 \vee d_j = 5 \vee d_j = 7\}$$

and $card(N^{dia}) = \sum_{i=1}^{N} card(N_i^{dia})$ as $N_i^{dia} \cap N_j^{dia} = \phi \;\; for\, i \neq j$.

### 3.2.2 Region- based Features

The region-based features were proposed in order to measure the plain, concave and convex regions and this variability of writing could be used for case and legibility of handwriting [LI93]. The region-based features used in the system are dominant points in the contours and direction primitives between dominant points. Prior to the process of finding dominant points, a Gaussian Average Filter is used to reduce the influence of digitisation noise. The filtered version of $A_i$ is denoted as:

$$\overline{A}_i = \{\overline{a}_i \mid i = 2,3,\ldots,M_i - 1\}.$$

After performing Gaussian Average Filter on $A_i$, each contour $C_i$ can be partitioned into a sequence of convex, concave and plain regions.

$$C_i = \bigcup_{j=1}^{T_i} R_{ij}^k$$

Where

$T_i$ is the number of disjointed regions of $C_i$

$R_{ij}^k, k \in \{\ 1,2,3\ \}$, are series of consecutive points on contours $C_i$, in such a way that :

$R_{ij}^1 = \{\ p_l \in C_i \mid p_l \text{ are consecutive points}, \overline{a}_l = 180\ \}$ (Plain region)

$R_{ij}^2 = \{\ p_l \in C_i \mid p_l \text{ are consecutive points}, \overline{a}_l < 180\ \}$ (Concave region)

$$R_{ij}^{3}=\{\ p_l \in C_i \mid p_l \text{ are consecutive points}, \bar{a}_l > 180\ \} \qquad \text{(Convex regions)}$$

Figures 3-3, 3-4, 3-5 and 3-6 show an example of a typical word with its concave, convex and plain regions.

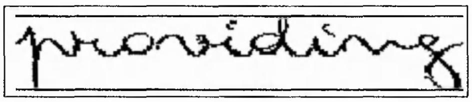


**Figure 3-3:** A typical word.

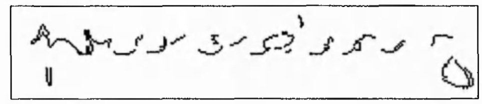


**Figure 3-4:** Concave regions.

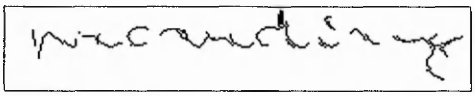


**Figure 3-5:** Convex regions.

**Figure 3-6:** Plain regions.

The contour angle $v_l$ at $p_l$ is defined within a support region and its value estimated by averaging angles $a_{lk}$, where $k = 1,2,3,....,K$ and $a_{lk}$ is formed by the pair of vectors $d_{l-k}$ and $d_{l+k-1}$. Denoting the sequence of contour angles in the region as;

V = $v_2 v_3 .... v_{M_i - 1}$, one can easily obtain the maximum within a convex region and the minimum in a concave region. All such maxima and minima constitute the local extremes of the curvature (corner points) along a word. More details of the above technique can be found in [LI93]. Figure 3-7 shows the corner points, which are detected on words after using Average Gaussian Filtering, with 2 iterations while $K = 3$ is considered. It should be noted that the experimental results show that as the number of iterations is increased then the effects of the filtering process will remove some of the dominant points as well as the noise. On the other hand if the number of iterations is not enough the system will detect some of the noise as dominant points.

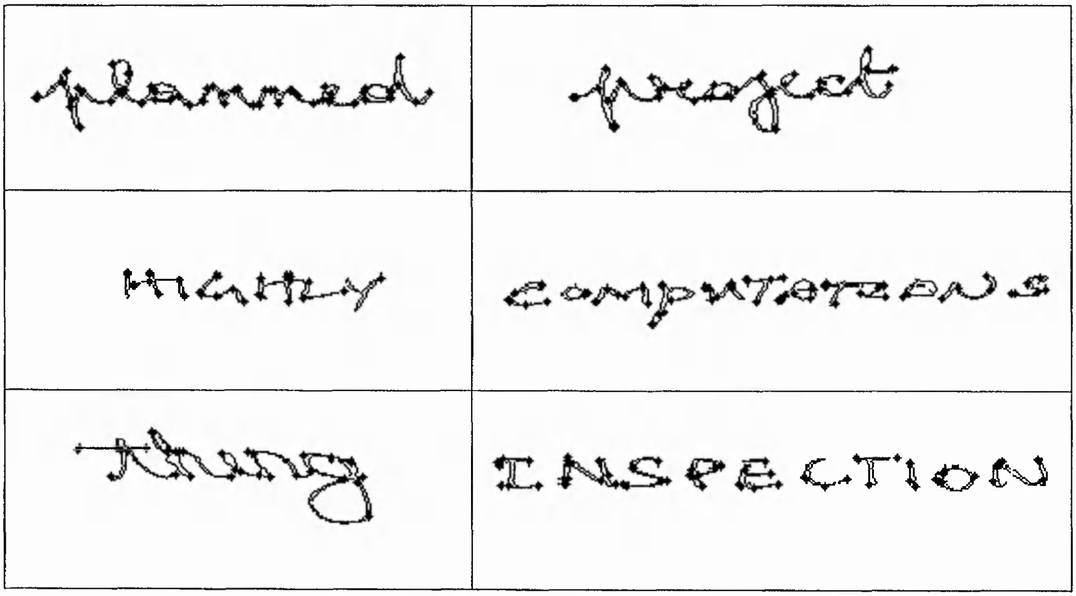


**Figure 3-7:** The detected dominant points on words.

Denoting $C_i^{cr} = \{\ p_j^{cr} \in C_i \mid j = 1,2,..,S_i\ \}$ as the dominant or critical points of the $i^{th}$ contour and $D_i^{cr} = \{\ d_j^{cr} \mid j = 1,2,...,S_i - 1\ \}$ as the direction primitives between dominant points, the region-based features are defined as follows:

*(1) Average Region Length* (**AREL**):

$$\text{AREL} = \frac{card(P)}{\sum_{i=1}^{N} \sum_{\substack{j=1 \\ k\in\{1,2,3\}}}^{T_i} card(R_{ij}^{k})}$$

*(2) Average Plain Region Length* (**APRL**):

$$\text{APRL} = \frac{card(P)}{\sum_{i=1}^{N} \sum_{j=1}^{T_i} card(R_{ij}^{1})}$$

*(3) Average Concave Region Length* (**ACAL**):

$$\mathbf{ACAL} = \frac{card(P)}{\sum_{i=1}^{N}\sum_{j=1}^{T_i} card(R_{ij}^2)}$$

*(4) Average Convex Region Length* (**ACVL**):

$$\mathbf{ACVL} = \frac{card(P)}{\sum_{i=1}^{N}\sum_{j=1}^{T_i} card(R_{ij}^3)}$$

*(5) Ratio of Sharp Angle of critical points to the total number of critical points* (**RSCR**):

$$\mathbf{RSCR} = \frac{\sum_{i=1}^{N} card(V_i^{cr,90})}{\sum_{i=1}^{N} card(C_i^{cr})}$$

Where

$$V_i^{cr,\theta} = \{v_j \in V_i \mid v_j < \theta, P_j \in C_i^{cr}, j = 2,3,....,M_i - 1\}$$

*(6) Ratio of filtered Sharp Angle to the total number of Points* (**RFSP**):

$$\mathbf{RFSP} = \frac{\sum_{i=1}^{N} card(\overline{A}_i^{90})}{card(P)}$$

Where

$$\overline{\mathrm{A}}_i^{\theta} = \{\overline{a}_j \in \overline{A}_i \mid \overline{a}_j < \theta , j = 2,3,..., M_{i-1}\} .$$

*(7) Ratio of critical vertical code to the total critical chain code* (**RVF**):

$$\mathbf{RVF} = \frac{card(\overline{N}^{ver})}{\sum_{i}^{N} card(C_i^{cr})}$$

Where

$$\overline{N}^{ver} = \bigcup_{i=1}^{N} \overline{N}_i^{ver}$$

$$\overline{N}_i^{ver} = \{\mathrm{d}_j^{cr} \in D_i^{cr} \mid d_j^{cr} = 2 \vee d_j^{cr} = 6\}$$

and $card(\overline{N}^{ver}) = \sum_{i=1}^{N} card(\overline{N}_i^{ver})$ as $N_i^{ver} \cap N_j^{ver} = \phi \quad for\ i \neq j$.

*(8) Ratio of critical horizontal code to the total critical chain code* (**RHF**):

$$\textbf{RHF} = \frac{card(\overline{N}^{hor})}{\sum_{i}^{N} card(C_i^{cr})}$$

Where

$$\overline{N}^{hor} = \bigcup_{i=1}^{N} \overline{N}_i^{hor}$$

$$\overline{N}_i^{hor} = \{\mathrm{d}_j^{cr} \in D_i^{cr} \mid d_j^{cr} = 0 \vee d_j^{cr} = 4\}$$

and $card(\overline{N}^{hor}) = \sum_{i=1}^{N} card(\overline{N}_i^{hor})$ as $N_i^{hor} \cap N_j^{hor} = \phi \quad for\ i \neq j$.

*(9) Ratio of critical diagonal to the total critical chain code* (**RDF**):

$$\textbf{RDF} = \frac{card(\overline{N}^{dia})}{\sum_{i}^{N} card(C_i^{cr})}$$

Where

$$\overline{N}^{dia} = \bigcup_{i=1}^{N} \overline{N}_i^{dia}$$

$$\overline{N}_i^{dia} = \{\mathrm{d}_j^{cr} \in D_i^{cr} \mid d_j^{cr} = 1 \vee d_j^{cr} = 3 \vee d_j^{cr} = 5 \vee d_j^{cr} = 7\}$$

and $card(\overline{N}^{dia}) = \sum_{i=1}^{N} card(\overline{N}_i^{dia})$ as $N_i^{dia} \cap N_j^{dia} = \phi \quad for\ i \neq j$

### 3.2.3 Windows-based Features

Any word image can be subdivided into 3 horizontal regions of interest corresponding to the upper, main and lower body of an image (Figure 3-8). The width of the upper, main and lower bodies is respectively 25%, 50% and 25% of the word height (distance between upper and lower base lines).

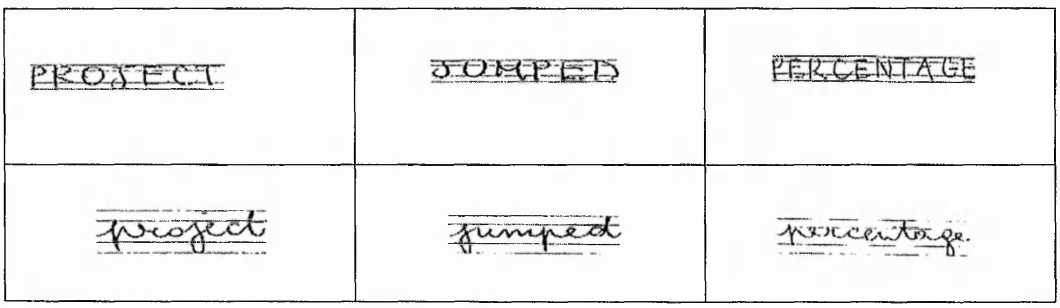


**Figure 3-8:** Three regions of interest within a window for some different word case samples.

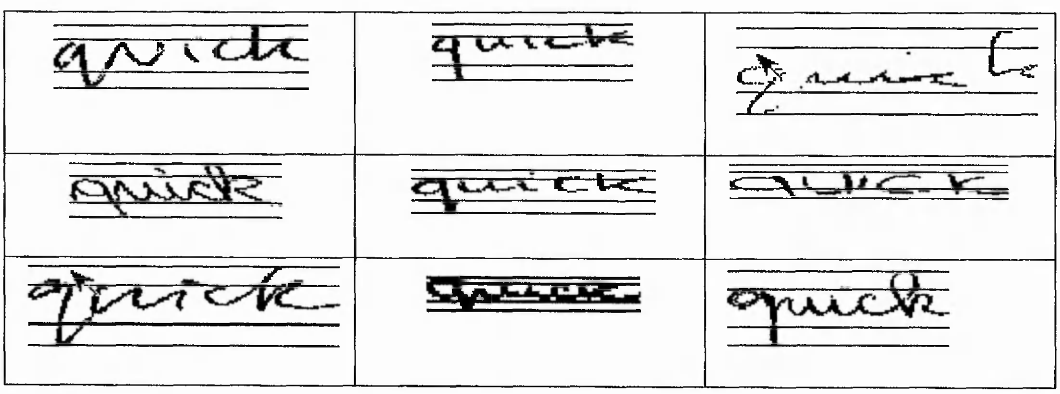


**Figure 3-9:** Three regions of interest within a window for some different styles of handwriting (one specific word).

As figure 3-8 shows, the number of pixels and the value of slope in each window should be different for uppercase, lowercase and mixed case word images. Figure 3-9 also shows how handwriting from different people could be different in each window. The following features were introduced to investigate this style characteristic.

Four values of slope, corresponding to the angle of a direction with the horizontal, are extracted from the 8 directions given by the Freeman code. The 4 values correspond to angles of 0, 45, 90 and 135 degrees respectively to the horizontal (Figure 3-10).

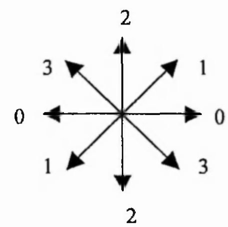



**Figure 3-10:** Representation of the four directions (slopes).

For a given window $i$ and a given slope $k$, the $\text{pointszone}(i \mid k)$ is computed as follows:

$$\text{pointszone}(i \mid k) = \frac{\left(\dfrac{card(i \mid k)}{\sum_k card(i \mid k)}\right)}{\max_{i,k}\left(\dfrac{card(i \mid k)}{\sum_k card(i \mid k)}\right)}$$

Where

$card(i \mid k)$ is the number of contour points with a given slope k

The total number of local features extracted for a given window position is a made up of 3 slope features for each of the 3 zones. These are defined as follows:

*(1) Ratio of vertical directions in lower window* (**RVLZ**):

$$\textbf{RVLZ} = \text{pontszone}(0 \mid 2)$$

*(2) Ratio of horizontal directions in lower window* (**RHLZ**):

$$\textbf{RHLZ} = \text{pointszone}(0 \mid 0)$$

*(3) Ratio of diagonal directions in lower window* (**RDLZ**):

$$\textbf{RDLZ} = \text{pointszone}(0 \mid 1) + \text{pointszone}(0 \mid 3)$$

*(4) Ratio of vertical directions in middle window* (**RVZM**):

$$\textbf{RVZM} = \text{pointszone}(1 \mid 2)$$

*(5) Ratio of horizontal directions in middle window* (**RHZM**):

$$\textbf{RHZM} = \text{pointszone}(1 \mid 0) + \text{pointszone}(1,4)$$

*(6) Ratio of diagonal directions in middle window* (**RDZM**):

$$\textbf{RDZM} = \text{pointszone}(1 \mid 1) + \text{pointszone}(1 \mid 3)$$

*(7) Ratio of vertical directions in upper window* (**RVZU**):

$$\textbf{RVZU} = \text{pointszone}(2 \mid 2)$$

*(8) Ratio of horizontal directions in upper window* (**RHZU**):

$$\textbf{RHZU} = \text{pointszone}(2 \mid 0)$$

*(9) Ratio of diagonal directions in upper window* (**RDZU**):

$$\textbf{RDZU} = \text{pointszone}(2 \mid 1) + \text{pointszone}(2 \mid 3)$$

In addition to the above features the following feature is also defined:

*(10) Ratio of number of points in middle area to total number of points* (**RPCE**):

$$\textbf{RPCE} = \frac{cardMid(P)}{card(P)}$$

Where

$cardMid(P)$ is the number of points in the middle zone.

### 3.2.4 Feature-Based Moments

In addition to the slope features described above, an additional feature, **NOM1**, based on the second moment is also extracted. The moment features capture the global information of word images, which could help in both case and legibility classification of handwriting [LONCARIC98].

$$M_1 = (\mu_{20} - \mu_{02})^2 + 4\mu_{11}^2$$

Where the co-ordinates of a contour pixel is given by the 2D binary image of the cursive word and the central moment is given by:

$$\mu_{pq} = \frac{1}{N}\sum_{i=1}^{N}(x_i - \bar{x})^p (y_i - \bar{y})^q$$

Where

$p_i = (x_i, y_i) \in P$ and,

$$\bar{x} = \frac{1}{N}\sum x_i;\ \bar{y} = \frac{1}{N}\sum y_i$$

and N is the total number of points in the contour word image.

### 3.2.5 Zero-Crossing Feature

As figure 3-11 shows the number of intersections of a horizontal line passing through the midline of a word are different. The following features were therefore introduced to make use of this characteristic. A horizontal line is drawn through the centre of the word.

$$\text{Centre of the word} = \frac{1}{S}\left(\sum_{i=1}^{S} x_i, \sum_{i=1}^{S} y_i\right)$$

Where

$S$ is the total number of points in the contour word images.

The number of intersections of this line with the contoured word gives the number of zero crossing (**NCRS**) (Figure 3-11).

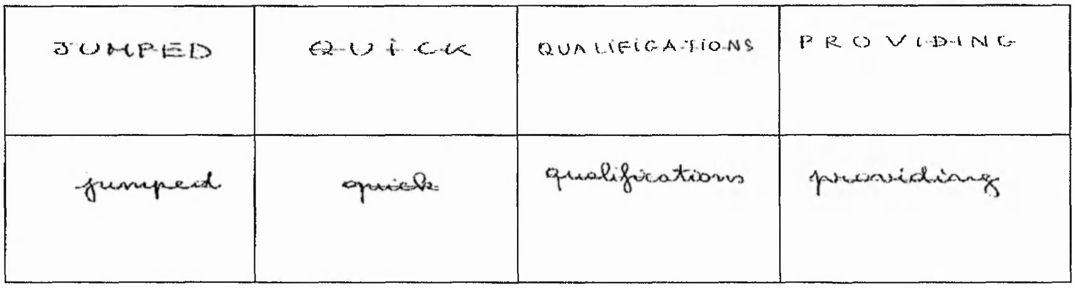


**Figure 3-11:** Horizontal lines are drawn from the centre of each word.

## 3.3 Group-based Features

To avoid using any segmentation technique, which may lead to errors, group-based features are introduced to deal with mixed case words [EBADIAN00]. First we need a definition of groups.

### 3.3.1 Group Definition

A group can be described as a sequence of connected pixels in a word image.

$$WI = \{G_i \mid G_i \cap G_j = \phi, i \neq j, i = 1,2,....,N \;, j = 1,2,....,N\} \text{ or}$$

$$WI = \bigcup_{i=1}^{N} G_i$$

Where $G_i = \{p_i \mid i = 1,2,...,N_i \;\&\; p_i = (\mathrm{x}_i, y_i)\}$.

$N$ is the number of groups in a word and $N_i$ is number of pixels in $i^{th}$ group of each word.

The group features used in our system are mainly based on:

*(a) Zoning information* [POWALKA95].

The zoning lines of the word image are the four lines that partition the word into three disjoint horizontal slices or zones. The width of the upper and lower zone is 25% of the word height and the width of the middle zone is 50% of the word height. $Y_L$ and $Y_U$ are horizontal lines at the of top and bottom of a word (Figure 3-12).

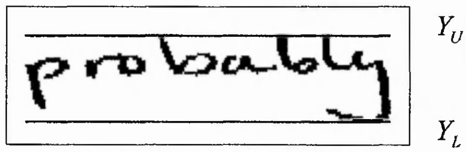



**Figure 3-12:** Upper and lower zone.

(b) *Bounding box of each group.*

A bounding box is a rectangular shape constructed of four points $P_{mima,i}, P_{mimi,i}, P_{mama,i}$ and $P_{mami,i}$ (figure 3-13). That denote the intersections between four lines; two horizontal line passing through the $Y_{\min,i}, Y_{\max,i}$ positions and two vertical lines passing through the $X_{\min,i}, X_{\max,i}$ positions. $Y_{\min,i}, Y_{\max,i}$ denote the minimum and maximum value of $y_i$ and $X_{\min,i}, X_{\max,i}$ denote the minimum and maximum value of $x_i$ for each pixel in $i^{th}$ group respectively.

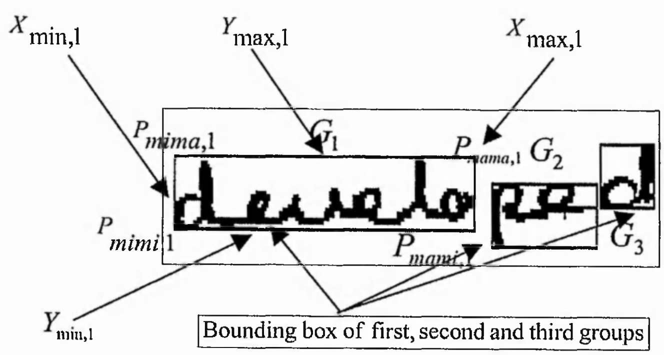



**Figure 3-13:** Groups and their bounding box.

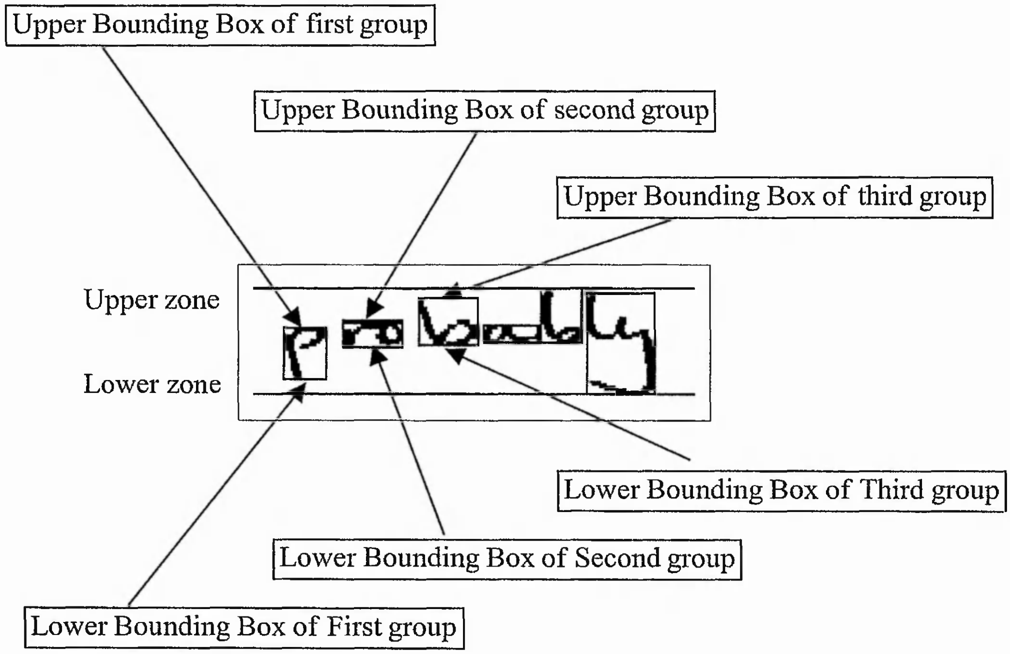



**Figure 3-14:** Illustration of group-based features.

## 3.4 Horizontal-based Histogram Features

Different characteristics of horizontal histograms were examined specifically to deal with mixed case words from writers who wrote purely cursively (Figure 3-15, 3-16 and 3-17). Figure 3-18 shows how the horizontal histogram of handwriting could vary from one person to another. The mean value of the columns on the horizontal histogram are calculated by:

$m = \frac{\sum_{i=1}^{n} col_i}{n}$ Where $col_i$ is number of black pixels in $i^{th}$ column of horizontal histogram and $n$ is number of columns in histogram.

*(1) Spread or first moment of the histograms* **(FMH):**

$$\text{FMH} = \frac{\sum_{i=1}^{n} |col_i - m|}{mn}$$

### 3.3.2. Group-based Features

The following group-based features are used in our system based on the above definition of groups. Since the first few letters in a word hold the most reliable information, only the first three groups in a word image are considered [ZHOA95]. Furthermore our experimental results show that increasing the number of groups is not beneficial and can lead to confusion due to the existence of ascenders or descenders in different positions of each word. Therefore the following features were extracted from the first three groups of each word.

*(1)* Number of groups in each word (N). Total number of groups in a word.

Since the first few letters in a word hold more reliable information, only features present in the first three groups in a word images are considered [ZHOA95].

*(2)* Ratios of distance between upper bounding line and upper zone line to distance between lower and upper zone line for the first three groups of the word (Figure 3-14).

$$RDUU_i = \frac{Y_U - Y_{\max,i}}{Y_U - Y_L}\ ,\quad i = \{1,2,3\}$$

*(3)* Ratios of distance between lower bounding line and lower zone line to distance between lower and upper zone line for the first three groups of the word (figure 3-14).

$$RDLL_i = \frac{Y_L - Y_{\min,i}}{Y_U - Y_L}, \{i = 1,2,3\}$$

*(2) The distance of the average height of columns* **(AH)**:

$$AH = \frac{K_2 - K_1}{n} \quad \text{Where } K_1 = \min\left\{k_i : k_i = \frac{col_i + col_{i+1}}{2} > m, i = 1,2,\ldots,n-1\right\}$$

$$K_2 = \min\left\{k_i : k_i = \frac{col_i + col_{i-1}}{2} > m, i = 2,\ldots,n\right\}$$

*(3) Ratio of number of black pixels in upper zone to number of black pixels in all three zone of a word.*

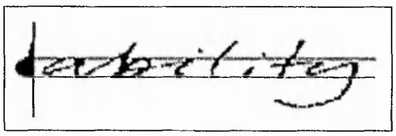


**Figure 3-15:** Horizontal histogram for lower case word.

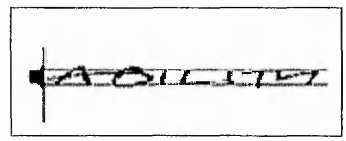


**Figure 3-16:** Horizontal histogram for uppercase word.

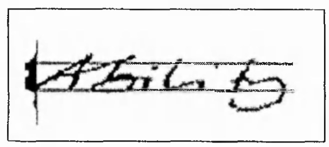


**Figure 3-17:** Horizontal histogram for mixed case word.

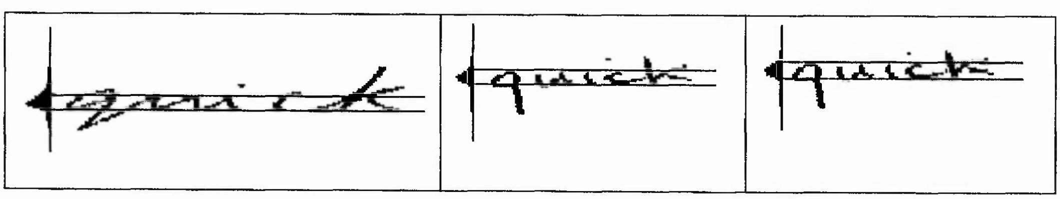


**Figure 3-18:** Horizontal histogram for different style of writing.

All the features used in this research, are numbered in table 3-2 for reference.

| | |
|---|---|
| 1 | Average Region Length |
| 2 | Average Concave Region Length |
| 3 | Average Plain Region Length |
| 4 | Average Convex Region Length |
| 5 | Ratio of original Sharp Angle to the total number of Points |
| 6 | Ratio of filtered Sharp Angle to the total number of Points |
| 7 | Ratio of critical vertical code to the total critical chain code |
| 8 | Ratio of critical horizontal code to the total critical chain code |
| 9 | Ratio of critical diagonal to the total critical chain code |
| 10 | Ratio of sharp angle of critical points to the total number of critical points |
| 11 | An estimate of the component length (disjoint contours) or averaged component $(C_i)$ length |
| 12 | Ratio of vertical direction to the total original chain code |
| 13 | Ratio of horizontal direction to the total original chain code |
| 14 | Ratio of diagonal direction to the total critical chain code |
| 15 | Ratio of vertical directions in lower window |
| 16 | Ratio of horizontal directions in lower window |
| 17 | Ratio of diagonal directions in lower window |
| 18 | Ratio of vertical directions in middle window |
| 19 | Ratio of horizontal directions in middle window |
| 20 | Ratio of diagonal directions in middle window |
| 21 | Ratio of vertical directions in upper window |
| 22 | Ratio of horizontal directions in upper window |
| 23 | Ratio of diagonal directions in upper window |
| 24 | Ratio of number of points in middle area to total number of points |
| 25 | Zero crossing |
| 26 | First moment feature |
| 27 | The distance of average height of columns |
| 28 | Ratio of number of black pixels in the upper zone to number of black pixels in all three zone of a word. |
| 29 | Spread or first moment of the histograms |
| 30 | Number of groups in each word |
| 31 | Ratios of distance between upper bounding box and upper zone to distance between lower and upper zone for the first groups of the word |
| 32 | Ratios of distance between upper bounding box and upper zone to distance between lower and upper zone for the second groups of the word |
| 33 | Ratios of distance between upper bounding box and upper zone to distance between lower and upper zone for the third groups of the word |
| 34 | Ratios of distance between lower bounding box and lower zone to distance between lower and upper zone for the first groups of the word |
| 35 | Ratios of distance between lower bounding box and lower zone to distance between lower and upper zone for the second groups of the word |
| 36 | Ratios of distance between lower bounding box and lower zone to distance between lower and upper zone for the third groups of the word |

**Table 3-2:** Thirty-six extracted features.

## 3.5 Summary

In this chapter thirty-six features are proposed for extraction from a word image in order to perform case and legibility classification. These features are contour based features, global features, region-based features, window-based features, features based on moments, features based on zero crossing, group-based features and horizontal histogram features. It should be pointed out that the aim of this section was to extract the different characteristics of each word image by introducing as many features as possible. The contribution of each feature for case and legibility classification will be assessed and, consequently, will be justified using a feature selection scheme (MDA) in the next chapter. In the next chapter we also show how these features can be used with both a Multiple Discriminant Analysis (MDA) and Probabilistic Neural Network (PNN) classifier to perform both case and legibility classification.

# 4. CLASSIFICATION METHODS

## 4.1 Introduction

In any classification method the main aim is to find patterns in the data, which can be used to discriminate between subgroups of the data and to identify important distinguishing factors. Recognition or classification may consist of one of the following tasks: 1) supervised classification (discriminant analysis) in which the input pattern is identified as a member of a predefined class, 2) unsupervised classification (clustering) in which the pattern is assigned to an unknown class.

There are many classification techniques in the literature such as linear or non-linear discriminant analysis, kernel-based classifier and k-nearest neighbourhood classifier [WEB99][PARZEN62]. Depending on the information available about the class-conditional densities, various strategies are utilized to design a classifier. If all class-conditional densities are completely specified, then the optimal Bayes decision rule can be used to design a classifier. However, class-conditional densities are usually not known in practice and must be learnt from the available training patterns. If the form of the class-conditional densities is known (e.g. multivariate Gaussian), but some of the parameters of the densities (e.g. mean vector and covariance matrices) are unknown, then we have a parametric decision problem. A common strategy for this kind of problem is to replace the unknown parameters in the density functions by their estimated values. If the form of class-conditional density is not known, then we operate in a nonparametric mode. In this case we must either estimate the density function (e.g. using a parzen window approach) or directly construct the decision boundary based on the training data (e.g Multiple discriminant analysis (MDA) and k-nearest neighbour rule).

In this equation, $W_b$ is the between-class scatter matrix, $W_w$ is the within-class scatter matrix and $\phi$ is the transformation we are searching for in order to form the optimal discrminant space. We can define the following, with $\underline{f}^{i,j} = \left(f_1^{i,j},\ldots,f_p^{i,j}\right)$ being the $p$ extracted features of word image $i$ in $j^{th}$ class and $n_j$ being the number of word images in class $j$ :

$$\overline{\underline{f}}^j = \frac{1}{n_j}\sum_{m=1}^{n_j}\underline{f}^{m,j} \text{ (Mean of features in } j^{th} \text{ class)} \qquad \text{Eq. (4-2)}$$

$$\overline{\underline{f}} = \frac{1}{n}\sum_{j=1}^{n} n_j \overline{\underline{f}}^j \text{ (Mean of features in all classes)} \qquad \text{Eq. (4-3)}$$

where $n$ is a number of classes $(j = 1,2,\ldots,n)$.

The within-class and between-class scatter matrices can be derived as follows:

$$W^j = \sum_{i=1}^{n_j}\left(\underline{f}^{i,j} - \overline{\underline{f}}^j\right)\left(\underline{f}^{i,j} - \overline{\underline{f}}^j\right)^t \text{ (covariance in } j^{th} \text{ class)} \qquad \text{Eq. (4-4)}$$

$$W_w = \sum_{j=1}^{n} W^j \text{ (Within class covariance)} \qquad \text{Eq. (4-5)}$$

$$W_b = \sum_{j=1}^{n} n_j\left(\overline{\underline{f}}^j - \overline{\underline{f}}\right)\left(\overline{\underline{f}}^j - \overline{\underline{f}}\right)^t \text{ (between class covariance)} \qquad \text{Eq. (4-6)}$$

Both the within-class scatters $W_w$ and the between-class scatter $W_b$ are analogous to their respective covariance matrices.

In looking for $\phi$ we can define

$$\underline{y} = \phi^t \underline{f} \text{ (Transform } \underline{f} \text{ by } \phi^t\text{)} \qquad \text{Eq. (4-7)}$$

$$\psi^j \equiv \left\{\underline{y}^j \mid \underline{f}^j \in j^{th}\ class, \underline{y}^j = \phi^t \underline{f}^j\right\}$$

$$\overline{\underline{y}}^j = \frac{1}{n_j}\sum_{\underline{y}\in\psi^j}\underline{y} \text{ (Mean of transformed features in } j^{th} \text{ class)} \qquad \text{Eq. (4-8)}$$

$$\overline{\underline{y}} = \frac{1}{n}\sum_{j=1}^{n} n_j \overline{\underline{y}}^j \text{ (Mean of transformed features in all classes)} \qquad \text{Eq. (4-9)}$$

In practice the choice of a classifier is a difficult problem and it is often based on which classifier is available or best known by the user [ROSEMARY97] [JAIN00]. In this research two approaches are used to classify the style of handwriting; Multiple Discriminant Analysis (MDA) and Probabilistic Neural Network (PNN) based on Parzen models. Based on its strengths in dealing with most complex distribution the PNN method provides a good candidate classification method. The PNN method assumes knowledge of the underlying class conditional probability density function. This density function is estimated from a training set (set of correctly classified samples) using Parzen models (see section 4.3.1). The following sections describe both classifiers in detail. The MDA develops a set of decision rules that uses the data to estimate the decision boundaries directly without explicit calculation of the probabilistic density functions. This discriminant space can be divided into as many regions as there are classes. The decision boundary between them can be used to assign an unknown word image to a class. In MDA the decision boundaries are linear.

## 4.2 Linear Classification - Multiple Discriminant Analysis Method

A linear discriminant transformation, Multiple Discriminant Analysis (MDA), is used to transform the feature space of 36 dimensions into an optimal discriminant space for a nearest mean classifier. A brief summary of the technique is given here for clarity, but for more detail see [RIPLEY 97].

The aim of MDA is to maximise the ratio of between-class variance and within-class variance:

$$\frac{|\tilde{W}_b|}{|\tilde{W}_w|} = \frac{|\phi' W_b \phi|}{|\phi' W_w \phi|} \qquad \text{Eq. (4-1)}$$

$$\widetilde{W}_w = \sum_j \sum_{y \in \psi^j} (\underline{y} - \underline{\bar{y}}^j)(\underline{y} - \underline{\bar{y}}^j)^t$$ (Within-class covariance of transformed features) Eq. (4-10)

$$\widetilde{W}_b = \sum_j n_j (\underline{\bar{y}}^j - \underline{\bar{y}})(\underline{\bar{y}}^j - \underline{\bar{y}})^t$$ (Between class covariance transformed features) Eq. (4-11)

from these it follows that

$$\widetilde{W}_w = \phi^t W_w \phi \qquad \text{Eq. (4-12)}$$

$$\widetilde{W}_b = \phi^t W_b \phi \qquad \text{Eq. (4-13)}$$

Taking the determinant of a scatter matrix is equivalent to finding the product of the eigenvalues, which, in turn, corresponds to the product of the variance. As may be seen with reference to Eq. (4-1) by maximising this ratio, we are looking for a transform $\phi$ that maximizes the between-class variance with respect to the within-class variance. The solution of Eq. (4-1) can be shown [GONZALEZ93][REPLEY93] to correspond to the generalised eigenvectors of the following equation:

$$W_b \underline{\phi}_j = \lambda_j W_w \underline{\phi}_j \qquad \text{Eq. (4-14)}$$

where the vectors $\underline{\phi}_j$ then form the columns of the matrix $\phi$.

In addition, the individual dimensions of the discriminant space created by each eigenvector $\underline{\phi}_j$ are now ordered. The between-class variance in dimension $j$ is proportional to the eigenvalue $\lambda_j$. Assuming a constant within-class variance, the higher the between-class variance of a dimension, the better the discriminant capacity of that dimension.

One additional step can be taken is to scale all of the within-class variances to uniform size in the discriminant space. The variance in dimension $j$ can be computed as $\underline{\phi}'_j W_w \underline{\phi}_j$ and each dimension can be scaled by replacing $\underline{\phi}_j$ with

$$\hat{\underline{\phi}}_j = \frac{\underline{\phi}_j}{\sqrt{\underline{\phi}'_j W_w \underline{\phi}_j}} \qquad \text{Eq. (4-15)}$$

giving each new dimension uniform variance.

The decision as to whether the particular word image is allocated to one class or another is then based on measuring the Euclidean distance between its transform scores (created by the MDA) and the centroids of all the classes in the discriminant space (nearest mean classifier). The nearest mean classifier is very simple and robust. Each pattern class is represented by a single prototype, which is the mean vector of all training samples in that class. Further, this classifier does not require any user specific parameters.

## 4.3 Non-linear Classification PNN Method

Besides using a linear method to perform style classification, a statistical classification method based on a Bayesian rule decision can also be used to classify the style of an unseen word. The basic idea behind the Bayesian decision rule is to calculate the probability density functions of the features of the word images in each of the classes $\omega_i$. This can be done both for case classification ($i$ = U (upper), L (Lower) and M (Mixed)) and for legibility of handwriting. The probability that a particular set of features from word image $\underline{f} = (f_1, \ldots, f_{36})$ comes from class $\omega_i$ is denoted as:

$p(\omega_i \mid \underline{f})$ where,

$$p(\omega_i \mid \underline{f}) = \frac{p(\underline{f} \mid \omega_i)p(\omega_i)}{\sum_{j=1}^{C} p(\underline{f} \mid \omega_j)p(\omega_j)} \qquad \text{Eq. (4-16)}$$

and $C$ is number of classes. This equation requires knowledge of the class-conditional density. This can be achieved by using a parzen model [PARZEN62].

### 4.3.1 Parzen Method

The accuracy of the Bayesian decision in Eq. (4-16) depends on the accuracy with which the underlying class-conditional density is estimated. A Parzen model [PARZEN62] is a class of smooth and continuous Probability Density Function (PDF) estimators, which become progressively more representative of the true class-conditional density as the number of samples increases. The Parzen model uses weight functions $W(d)$ which has a maximum value at $d = 0$ and which decreases as the absolute value of $d$ increases. A general formulation of the Parzen model is described by:

$$g(\underline{f}) = \frac{1}{n_j \sigma_1 \cdots \sigma_p} \sum_{i=1}^{n_j} W\left(\frac{(f_1 - f_1^i)}{\sigma_1}, \ldots, \frac{(f_p - f_p^i)}{\sigma_p}\right) \qquad \text{Eq. (4-17)}$$

where $\underline{f}^i = (f^i{}_1, \ldots, f_p^i)$ and $p$ are the sample points (extracted features) and number of features in the training set, $\sigma_k$ is the variation of $k^{th}$ features ($k = 1,2,\ldots p$) of points that surround each sample in the training set, $n_j$ is the number of samples in class $\omega_j$, $W$ is the weight function and $f_k^i$ is the $k^{th}$ feature which is extracted from $i^{th}$ word image belonging to the $\omega_j$ class.

In general each Parzen method should have multiple $\sigma_i$ values. However to simplify the model a special case can be assumed where $\sigma = \sigma_1 = \sigma_2 = \ldots = \sigma_p$ for all of the weights of function $W$. A more general density estimator, which assumes a Gaussian kernel distribution, is used in this study, which is well behaved and easily computed. Thus Eq. (4-17) becomes:

$$g(\underline{f})=\frac{1}{n_j\sigma^p\sqrt{2\pi}}\sum_{i=1}^{n_j}e^{\frac{-\|\underline{f}-\underline{f}^i\|}{2\sigma^2}} \qquad \text{Eq. (4-18)}$$

As we do not know in advance which features are important and which are not the presence of features whose variation is meaningless has a dilutive effect on the useful features. We want the variation of unimportant features to be small so that they exert minimal influence on the distance measure computed between an unknown point (test word) and each member of the training case. The solution to this problem is to use a separate $\sigma$ weight for each feature. Eq.(4-18) then changes to: $g(\underline{f})=\frac{1}{\prod_{k=1}^{p}\sqrt{2\pi}\sigma_k}\sum_{i=1}^{n_j}e^{-D(\underline{f},\underline{f}^i)}$

Eq. (4-19)

where

$$D(\underline{f},\underline{f}^i)=\sum_{k=1}^{p}\left(\frac{f_k-f_k^i}{\sigma_k}\right)^2 \qquad \text{Eq. (4-20)}$$

In this experiment both approaches were tested in order to evaluate the effectiveness of each method. In characterising the function represented by Eq. (4-18) the estimation of $\sigma_i$ is critical [PARZEN62]. A good criterion for selecting appropriate values of $\sigma_i$ is the number of correctly classified cases that each value produces.

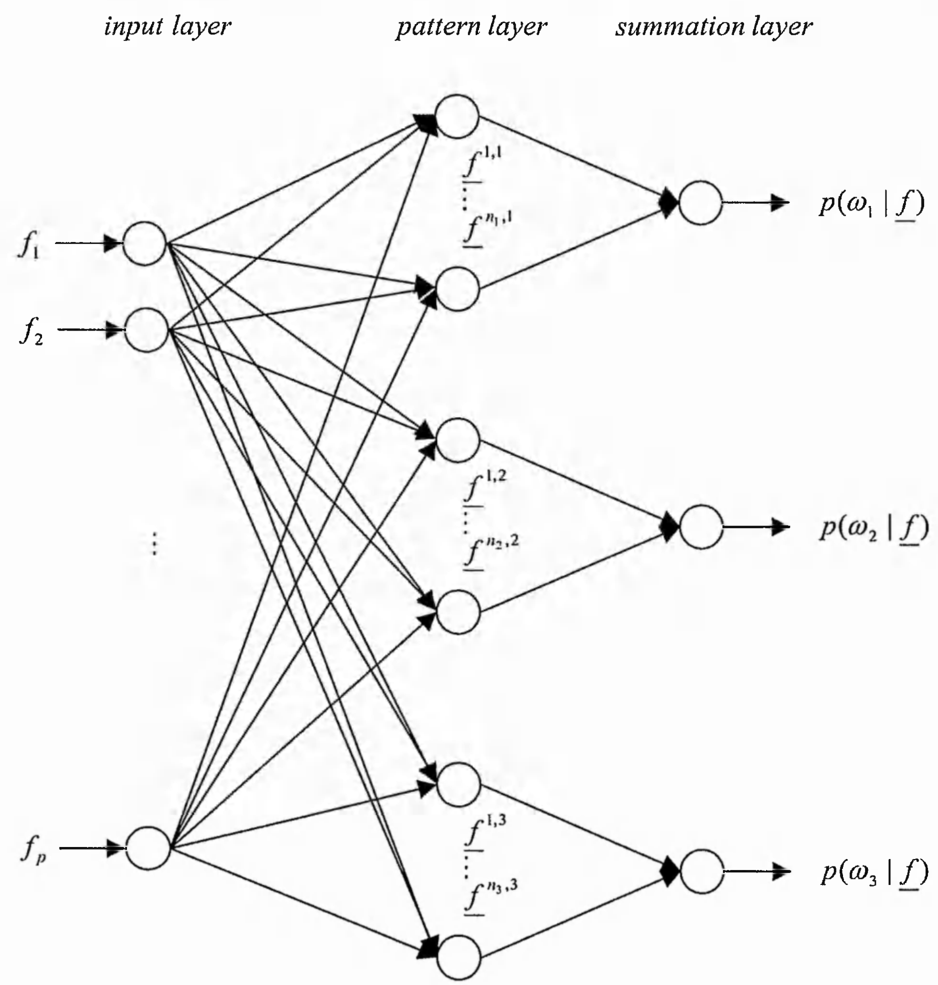
### 4.3.2 Optimising the $\sigma$

For each particular $\sigma$ a set of Parzen density estimators based on the training data set is estimated. The number of correctly classified words produced by each value is then used to judge the efficiency of a particular value of $\sigma$ . To estimate an unbiased correct classification rate for each $\sigma$ , a leave–one-out method was used. In this method, all of the training data set belonging to each class except one is used to train the system and the remaining datum is used for testing. This training and testing using the leave-one-out method was repeated until every datum element in the two or three different classes had been independently tested. The leave-one-out method thus gives class bounds of the true performance of the classifier [FUKUNAGA89].

The numbers of misclassified words for each $\sigma$ are then counted as an error function. A final value of $\sigma$ is then chosen that minimises the error function (number of misclassifications). The minimisation technique involves two stages. First a global search over a reasonable range is used to find a rough minimum. The range can be determined iteratively such that the error rate is minimised. Then a golden section method [RIPLEY97] is used to refine the estimate. Details were extensively reported by [SCHIOLER92][SPECHT91] and therefore are not reported here.

### 4.3.3 Probabilistic Neural Network

The non-parametric classifier described in the previous section can be implemented as a Probabilistic Neural Network structure. Figure 4-1 shows a neural network organization for classification of input pattern $\underline{f} = (f_1, \cdots, f_p)$ ($p$ indicates the number of features) into three classes. The input unit is simultaneously distributed to all neurons in the pattern layer.




$$g(\underline{f}) = \frac{1}{\prod_{k=1}^{p} \sqrt{2\pi}\sigma_k} \sum_{i=1}^{n_j} \exp\left(-D\left(\underline{f} - \underline{f}^i\right)\right)$$

**Figure 4-1:** Organization for classification of pattern into categories.

The network is trained by setting the $W_p$ weight vector in one of the pattern units equal to each $\underline{f} = (f_1, \cdots, f_p)$ pattern in the training set. The dot product of the input pattern vector $\underline{f}$ with a weight vector $W_p$ is calculated, which performs a non-linear operation on $Y_p = \underline{f}.W_p$ [DONALD90]. The summation units simply sum the inputs from the pattern units that correspond to the class from which the training pattern was selected and then a Bayes decision rule is used to calculate the probability density functions for each class.

Compared to traditional multi-layer perceptron (MLP) networks, our kernel-based method has a simple architecture consisting of two layers of weights, in which the first layer contains the parameters of the kernel functions and the second layer forms linear combinations of the activations of the kernel functions to generate the outputs. A MLP network often has many layers of weights and a complex pattern of connectivity. All the parameters in a MLP network are usually determined at the same time as part of a single global training strategy involving supervised training. Our kernel-based method, however, is typically trained in two stages, with the kernel functions being determined first using unsupervised techniques on the input data alone and then the second layer weights subsequently being found by fast linear supervised methods.

### 4.3.4 Comparison of Appropriate Classification Methods

Most of the standard statistical classification algorithms assume some knowledge of the distribution of the random variables used to classify. Specifically, a multivariate normal distribution is frequently assumed, and the training set is used only to estimate the mean vectors and covariance matrix of the populations. This means that large deviations from normalities usually causes a classifier to fail. Multimodal distributions cause even most nonparametric methods to fail. An advantage of neural networks is that they can typically handle even the most complex distributions. Multiple layer feed forward networks (MLFNs) have been shown to be robust classifiers. On the other hand, there are two main problems with MLFN: there is little knowledge about 1. how they operate and 2. what behaviour is theoretically expected of them. Another major problem with MLFN is that their training speed can be very slow. The PNN, however, usually trains orders of magnitude faster than

MLFNs, and classifies as well as or better than they do. Its main drawback is that MLFN is slow to classify. However, most important of all for many applications is that the PNN method can provide mathematically sound confidence levels for its decisions. This fact alone has made the PNN a favourite for our applications.

Another major advantage of using a PNN is the way it handles outliers; points that are very different from the majority. In fact, outliers will have no real impact on decisions regarding the more frequent cases, yet they will be properly handled if the data is valid. Existing outliers is an important issue for other neural network models or traditional statistical techniques since they can totally devastate the outcome.

As mentioned earlier, it should be emphasised that the outputs of our classifier also have a precise interpretation as the posterior probabilities of class membership. The ability to interpret outputs in this way is of central importance in the effective application of classifiers, as it may be used for rejecting a test pattern in case of doubt. Thus it would have some performance gains over other methods like k-nearest neighbour or support vector machine. Finally, the PNN technique is strongly based on Bayes's method of classification. This means that provided the true probability density function is known, there is a Bayes optimal decision rule that will minimise the expected cost of misclassification.

## 4.4 Feature Efficiency

In order to evaluate the efficiency of the PNN approach we investigated a means of minimising the PNN input layer without compromising the performance of the system. The multiple descriminant analysis (MDA) was applied to all 36 extracted features in this study in order to select the best $n$ features prior to training the PNN classifier for case classification and legibility of handwriting. In other words, MDA was applied on the set of 36 pre-recognition features to select those features that contribute the most to a discriminant between the pair of classes (upper/lower, upper/mixed, lower/mixed case words) and between all three classes (upper, lower and mixed case words). MDA was also applied on the set of 36 pre-recognition features to select those features that contribute the most to a discrimination between legible and non-legible handwriting words.

Features corresponding to the largest elements of the eigenvector, $\underline{\phi} = (\phi_1, \phi_2, ..., \phi_{36})$ (see Eq. 4-14), are then considered to be the best features for use in the PNN system [HEIJDEN95]. The percentage of contribution ($con$) of the selected feature sets is the ratio of the sum of coefficient $\phi_i$ that has been selected to the sum of total coefficients as described below:

$$con = \frac{\sum_{i \in S} \phi_i}{\sum_{j=1}^{36} \phi_j} \times 100 \qquad \text{Eq. (4-26)}$$

where S is set of selected features and $\phi_i$ is $i^{th}$ element (coefficient) of the eigenvector.

The effectiveness of each feature in a classification system for discrimination between each of the pairs of classes and all three classes is examined using:

$$con_i = \frac{\phi_i}{\sum_{j=1}^{36} \phi_j} \qquad \text{Eq. (4-27)}$$

Thus $con_i$ is a measure of the contribution of the $i^{th}$ feature.

### 4.4.1 Feature Efficiency in Case Classification

Table 4-1 shows the effectiveness of each feature for discrimination between each pair of class (lower/upper, upper/mixed, and lower/mixed case) and all three classes (lower/upper/mixed case). The first column denotes the feature number whilst the second, third and fourth columns of this table show the contribution of the selected features for classification between the two class lower/upper, upper/mixed, and lower/mixed case words and the fifth column for between the three class lower/upper/mixed case words respectively.

For this purpose the eigenvectors of the existing training set are calculated using the MDA. Then by using Eq. (4-12) and a threshold of 0.009 a set of best features are selected. Using this threshold more than 90% of the variation can be extracted for each pair of class and for all three classes.

| Feature's number | Upper/lower | Upper/mixed | Lower/mixed | Upper/lower/mixed case word images |
|---|---|---|---|---|
| 1 | 0.063801 | 0.08000 | 0.02694 | 0.07781 |
| 2 | 0.046666 | 0.02000 | 0.09969 | 0.00269 |
| 3 | 0.014985 | 0.04000 | 0.09746 | 0.03392 |
| 4 | 0.02477 | 0.04000 | 0.10398 | 0.02473 |
| 5 | 0.015494 | 0.00600 | 0.00738 | 0.01524 |
| 6 | 0.03625 | 0.00300 | 0.00215 | 0.02512 |
| 7 | 0.01743 | 0.14000 | 0.05045 | 0.07689 |
| 8 | 0.011272 | 0.14000 | 0.06475 | 0.07444 |
| 9 | 0.0150562 | 0.14000 | 0.05004 | 0.07608 |
| 10 | 0.111768 | 0.06000 | 0.00026 | 0.11308 |
| 11 | 0.053117 | 0.04000 | 0.03755 | 0.04255 |
| 12 | 0.069593 | 0.00100 | 0.05014 | 0.07595 |
| 13 | 0.108531 | 0.00200 | 0.10775 | 0.08650 |
| 14 | 0.040104 | 0.00700 | 0.00001 | 0.06101 |
| 15 | 0.003639 | 0.00200 | 0.00166 | 0.00295 |
| 16 | 0.004465 | 0.00500 | 0.00249 | 0.00635 |
| 17 | 0.002636 | 0.00050 | 0.00469 | 0.00085 |
| 18 | 0.003851 | 0.00200 | 0.00289 | 0.00509 |
| 19 | 0.000902 | 0.0008 | 0.00048 | 0.00044 |
| 20 | 0.001141 | 0.00100 | 0.00237 | 0.00061 |
| 21 | 0.010769 | 0.00800 | 0.00360 | 0.00973 |
| 22 | 0.00582 | 0.0001 | 0.00518 | 0.00216 |
| 23 | 0.000785 | 0.0016 | 0.00327 | 0.00255 |
| 24 | 0.021444 | 0.01400 | 0.00772 | 0.01870 |
| 25 | 0.177712 | 0.00000 | 0.04256 | 0.00531 |
| 26 | 0.11785 | 0.01010 | 0.00080 | 0.01662 |
| 27 | 0.024745 | 0.04670 | 0.08721 | 0.04551 |
| 28 | 0.030586 | 0.02770 | 0.00243 | 0.03430 |
| 29 | 0.033569 | 0.08640 | 0.03133 | 0.02259 |
| 30 | 0.010035 | 0.01060 | 0.01245 | 0.01068 |
| 31 | 0.007489 | 0.00750 | 0.00800 | 0.00038 |
| 32 | 0.006411 | 0.00360 | 0.02509 | 0.00087 |
| 33 | 0.001301 | 0.00030 | 0.00582 | 0.00047 |
| 34 | 0.006518 | 0.01720 | 0.01464 | 0.01544 |
| 35 | 0.001066 | 0.00750 | 0.00663 | 0.00304 |
| 36 | 0.004481 | 0.00810 | 0.01174 | 0.00933 |

**Table 4-1:** Effectiveness of each feature in classification between each pair of classes and three case classifications.

Figures 4-2, 4-3, 4-4 and 4-5 show the selected features corresponding to the largest eigenvector's elements for upper/lower, upper/mixed, lower/mixed and upper/lower/mixed case word images respectively. Using Eq. (4-26) it can easily be seen that 20, 15, 23 and 25 features contribute 93%, 94%, 98% and 98% of the variation for each of the respective classifications. The selected features for each two or three class classification are shown in table 4-2. For more detail on the selected features the reader is referred again to table 3-2 page 3-20.

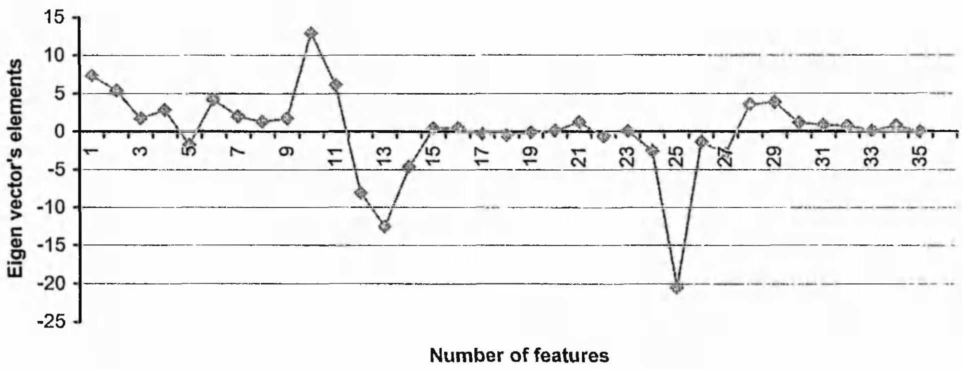



**Figure 4-2:** The 20 largest eigenvector weights capture (93%) of the variability between the lower and upper case word images.

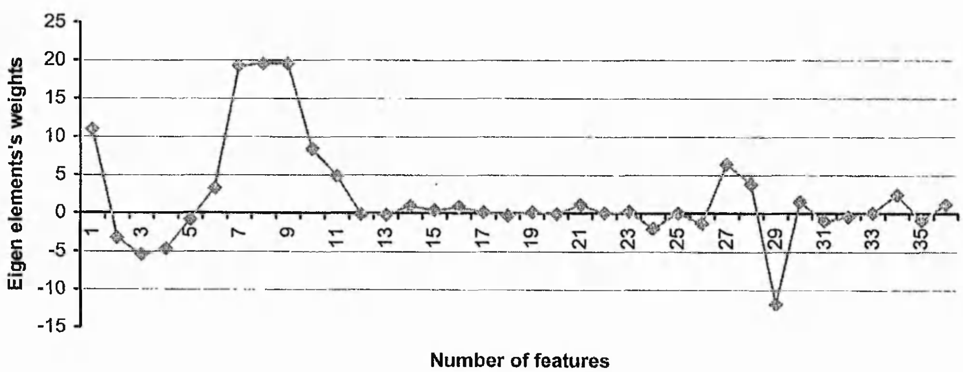



**Figure 4-3:** The 15 largest eigenvector weights capture (94%) of the variability between the upper and mixed case word images.

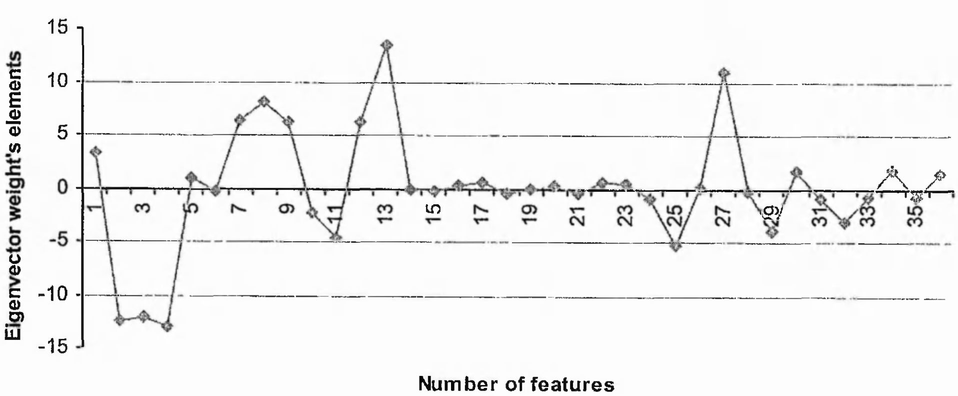



**Figure 4-4:** The 23 largest eigenvector weights capture (98%) of the variability between the lower and mixed case word images.

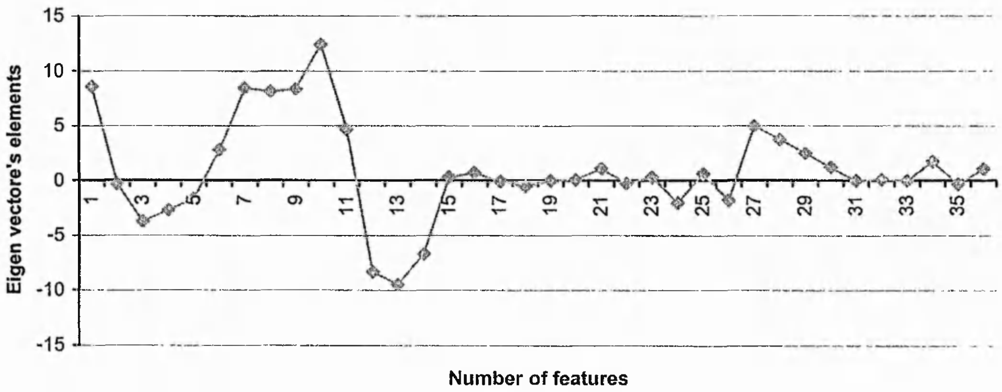



**Figure 4-5:** The 25 largest eigenvector weights capture (98%) of the variability between the lower, upper and mixed case word images.

| Case Classification | Selected Features |
|---|---|
| Upper-lower | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 24, 25, 26, 27, 28, 29 |
| Upper-mixed | 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 24, 27, 28, 29, 34 |
| Lower-mixed | 1, 2, 3, 4, 5, 7, 8, 9, 10, 11, 12, 13, 24, 25, 27, 29, 30, 31,32, 33, 34, 35, 36 |
| Lower-upper-mixed | 1, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 16, 18, 21, 24, 25, 26, 27, 28, 29, 30, 34, 36 |

**Table 4-2:** Selected features for classification between upper, lower and mixed case words images.

### 4.4.2 Feature Efficiency for Handwriting Legibility Classification

Table 4-3 shows the effectiveness of each feature in the classification system for each pair of classes (legible/illegible, legible/middle, illegible/middle writer) and for the three classes (legible/illegible/middle writer). This effectiveness is calculated using Eq. (4-27). The columns of this table show (in order) the feature number, the contribution of the selected features for classification between the legible/illegible, legible/middle, illegible/middle and legible/illegible/middle writer samples respectively.

| Feature's number | Legible/Illegible writer | Legible/Middle writer | Illegible/Middle writer | Legible/Illegible/ Middle writer |
|---|---|---|---|---|
| 1 | 0.01643 | 0.04220 | 0.00249 | 0.00220 |
| 2 | 0.01799 | 0.03556 | 0.08460 | 0.06536 |
| 3 | 0.11124 | 0.00347 | 0.11166 | 0.11711 |
| 4 | 0.10257 | 0.00406 | 0.10051 | 0.10896 |
| 5 | 0.01672 | 0.01036 | 0.00915 | 0.01375 |
| 6 | 0.00310 | 0.04615 | 0.00058 | 0.02279 |
| 7 | 0.12701 | 0.03176 | 0.12742 | 0.10398 |
| 8 | 0.11938 | 0.03622 | 0.12660 | 0.10500 |
| 9 | 0.12852 | 0.03391 | 0.12771 | 0.10329 |
| 10 | 0.06621 | 0.05373 | 0.01971 | 0.01245 |
| 11 | 0.14734 | 0.06645 | 0.20386 | 0.09133 |
| 12 | 0.00197 | 0.07639 | 0.01046 | 0.02784 |
| 13 | 0.00506 | 0.03276 | 0.01848 | 0.01799 |
| 14 | 0.01841 | 0.01804 | 0.00036 | 0.01135 |
| 15 | 0.00038 | 0.00204 | 0.00041 | 0.00144 |
| 16 | 0.00028 | 0.00002 | 0.00038 | 0.00165 |
| 17 | 0.00233 | 0.00089 | 0.00173 | 0.00227 |
| 18 | 0.00227 | 0.00192 | 0.00264 | 0.00061 |
| 19 | 0.00050 | 0.00145 | 0.00008 | 0.00018 |
| 20 | 0.00197 | 0.00002 | 0.00075 | 0.00207 |
| 21 | 0.00134 | 0.00199 | 0.00096 | 0.00167 |
| 22 | 0.00115 | 0.00222 | 0.00168 | 0.00040 |
| 23 | 0.00235 | 0.00122 | 0.00366 | 0.00282 |
| 24 | 0.00426 | 0.00411 | 0.00425 | 0.00712 |
| 25 | 0.00000 | 0.39834 | 0.00454 | 0.10427 |
| 26 | 0.01589 | 0.00806 | 0.01111 | 0.01417 |
| 27 | 0.02431 | 0.03460 | 0.00005 | 0.02060 |
| 28 | 0.01007 | 0.01331 | 0.00172 | 0.00632 |
| 29 | 0.01847 | 0.00630 | 0.00969 | 0.01250 |
| 30 | 0.00359 | 0.00327 | 0.00176 | 0.002331 |
| 31 | 0.00610 | 0.00920 | 0.00175 | 0.00251 |
| 32 | 0.00025 | 0.00100 | 0.00343 | 0.00179 |
| 33 | 0.11927 | 0.00827 | 0.00240 | 0.00718 |
| 34 | 0.00028 | 0.00254 | 0.00006 | 0.00003 |
| 35 | 0.00528 | 0.00212 | 0.00158 | 0.00149 |
| 36 | 0.00604 | 0.00278 | 0.00149 | 0.00405 |

**Table 4-3:** Effectiveness of each feature in classification between each pair of classes and three in legibility of handwriting.

Figures 4-6, 4-7, 4-8 and 4-9 show the selected features corresponding to the largest elements of the eigenvector, extracted from the training set for legibility of handwriting [HEIJDEN95]. For this purpose the eigenvectors of the existing files or training set are calculated then a threshold of 0.005 is used to select the set of best features. By using the threshold of 0.005 more than 90% of the variation can be extracted for every pair of classes or for all three classes. Using Eq. (4-26) it can again be seen that 20, 16, 15 and 13 features respectively contribute 97%, 95%, 93% and 96% of the variation between legible/illegible, legible/middle, illegible/middle writer and legible/illegible/middle writer. The selected features for each pair or three class classification are shown in table 4-5. For more detail on the selected features the reader is referred to table 3-2 page 3-20.

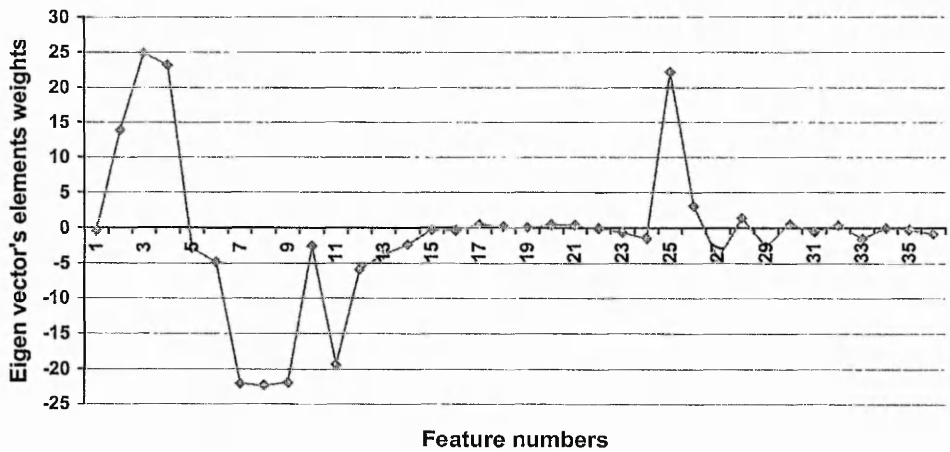



**Figure 4-6:** The 20 largest eigenvector weights capture (97%) of the variability between, legible, illegible and middle handwriting.

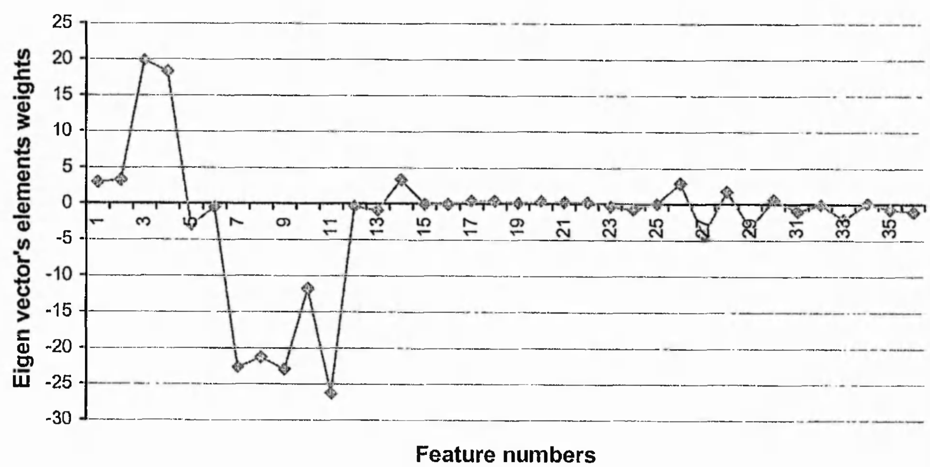



**Figure 4-7:** The 16 largest eigenvector weights capture (95%) of the variability between, legible and illegible handwriting.

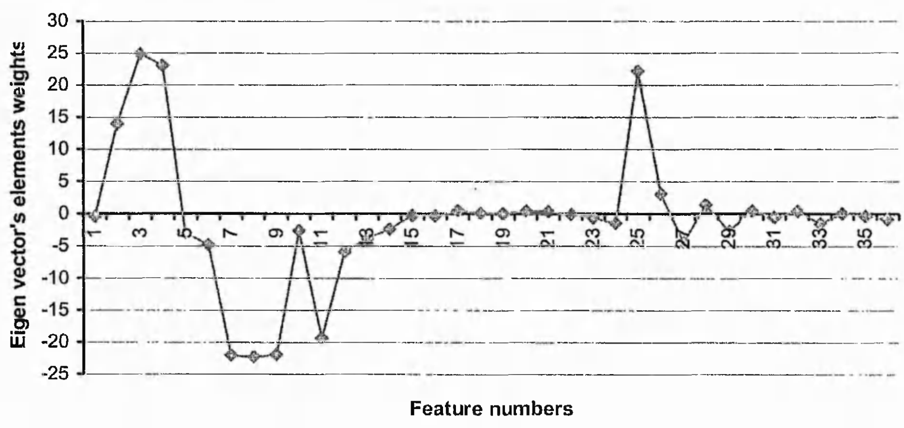



**Figure 4-8:** The 15 largest eigenvector weights capture (93%) of the variability between, legible and Middle handwriting.

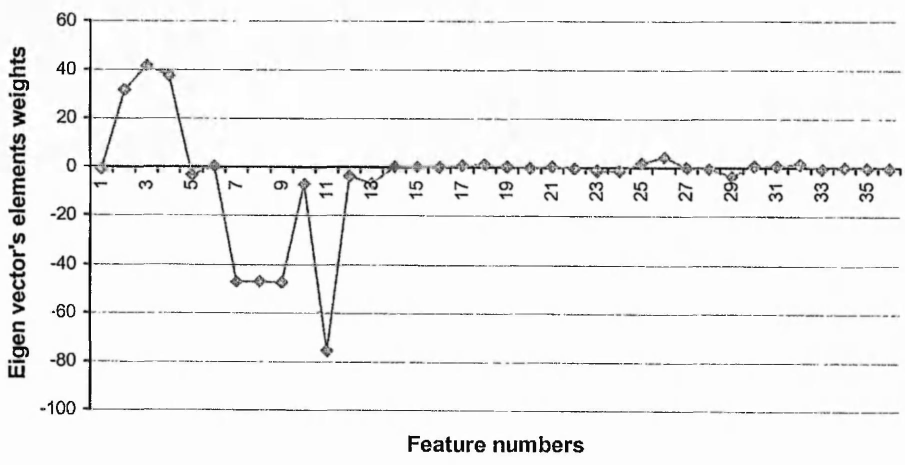



**Figure 4-9:** The 13 largest eigenvector weights capture (96%) of the variability between Middle and illegible handwriting.

| Legibility of writing | Selected features |
|---|---|
| Legible-Illegible writer | 1, 2, 3, 4, 5, 7, 8, 9, 10, 11, 14, 26, 27, 28, 29, 33 |
| Legible- Middle writer | 1, 2, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 25, 27, 28 |
| Middle- Illegible writer | 2, 3, 4, 5, 7, 8, 9, 10, 11, 12, 13, 26, 29 |
| Legible -Middle- Illegible writer | 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 24, 25, 26, 27, 28, 29, 33 |

**Table 4-4:** Selected features for classification between, legible, illegible and Middle handwriting.

## 4.5 Summary

In this chapter both linear and non-linear classification methods are explained for use in case and legibility of handwriting classification. In the Multiple Discriminant Analysis (MDA) method a nearest mean classifier is used to classify each new pattern. The MDA technique was also used to select the best features for each category of classification. In the non-linear classification method a Probability Neural Network (PNN) based on Bayesian decision is introduced to predict the legibility or case of an unknown handwriting sample. In the PNN approach, a Parzen's method of density estimations was used to estimate a class conditional density function from the available training data.

In the next two chapters we show how these classifications can be used for style of writing. Chapter 5 shows experiments for classification of word images into upper, lower and mixed case and chapter 6 shows experiments for legibility of writing.

# 5. CASE CLASSIFICATION

## 5.1 Introduction

One of the major difficulties in handwriting recognition is dealing with variability of style of handwriting. There are many ways that have been proposed to improve CSR performance (see chapter 2). Automatic case classification is one of the first steps in this general direction. The pre-classification of words into upper, lower and mixed cases would provide a useful means of reducing word ambiguity. If it were possible to classify the case of a word image prior to recognition, then the size of the lexicon used for any individual word recognition could be significantly reduced as only single case templates need be used. Such a system consumes less memory and computation resources and exhibits less confusion errors. Tim Kam Ho and Gorge Nagy [HO01] have already shown that identifying character types such as lowercase, uppercase, digit and punctuation or special characters make recognition much easier. Thus, in this chapter the relative performance of a Multiple Discriminant Analysis (MDA) [EBADIAN99a][EBADIAN99b][EBADIAN00] and a Probabilistic Neural Network (PNN) based on the Bayes function techniques are compared for the classification of off-line handwritten words into upper, lower and mixed case images. The two case classification techniques (PNN and MDA) were therefore applied on our existing data set, which consists of scanned images obtained from 9 writers each approximately containing 150 words at 200x100-dpi resolution (see Appendix A).

### 5.1.1 Handwriting data samples

The choice of a data set for collection is not trivial. The set should be designed as good representative for a large vocabulary. Previous work [JEDRZEJEWSKI97] has indicated the need for a careful choice of sample words to allow a good representation of a large vocabulary.

Kassel in 1995 [KASSEL95] has discussed the design aspects of such data sets and sample words used in this research were chosen based on that work. In his work a set of significant letter sequences is proposed first. This was done using a lexicon of approximately 33000 words. The set of letter sequence was enlarged by adding some additional sequences: all 26 characters in the words' initial positions 23 characters used as the word final position, 16 characters in the double form ("tt", "ll", etc.) and 15 letter pairs considered difficult to segment due to their similarity to some single letters ("rn", and "m", etc.). As a result significant character sequence is proposed in the following table 5-1.

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| ability | dd | izing | ol | squ | vu | #o | h# |
| able | de | ju | oo | ss | vv | #p | I# |
| ably | ding | ke | ously | st | wa | #q | k# |
| alized | ee | king | over | ta | work | #r | l# |
| an | equ | la | ow | ted | zzl | #s | m# |
| ar | es | lc | pa | ter | #a | #t | n# |
| ate | exp | ling | pe | th | #b | #u | o# |
| ations | form | lization | pl | tically | #c | #v | p# |
| back | fully | ln | po | ting | #d | #w | r# |
| bb | gg | lo | pp | tively | #e | #x | s# |
| bu | ha | ma | pro | tr | #f | #y | t# |
| cc | he | mb | qualify | tt | #g | #z | u# |
| ch | hing | ment | que | uff | #h | a# | w# |
| ci | ho | mi | quizzic | um | #I | b# | x# |
| cl | ification | nc | re | und | #j | c# | y# |
| comm | Ight | nn | ring | ur | #k | d# | z# |
| comp | ii | ography | rn | uv | #l | e# | |
| con | ingly | oi | rr | uzz | #m | f# | |
| ction | is | oj | sh | vi | #n | g# | |

**Table 5-1:** A set of significant character sequences (character '#' represents a word boundary). Adapted from[KASSEL95]

Finally a set of 12 sentences (table 5-2) has been designed and the words in the sentence have been chosen in such a way that full coverage of the significant letter sequence be achieved.

| |
|---|
| a quick brown fox has jumped over the lazy dog |
| providing the feedback attains its zero roots the project can theoretically be accurately planned |
| this is not to say that an ability to deal with generalized experimental formalism is not appropriate |
| a percentage of juvenile crime can now be foiled by newly developed cling stuff |
| qualifications in geography are commonly horrendously overrated |
| even relatively improbable suggestions and additions are to be fully kept and queued for inspection |
| visualization of quizzical equations can amazingly simplify the most puzzled computations |
| a daring article might probably question a working software construction |
| jump skiing is a particularly uncertain and vulnerable to hoax thing |
| few highly alcoholic long drinks will turn even a shabby xylophone into a superb jazz support |
| advertising bureau channels its capacity savvy and funds into tempting ambitious youngsters |
| automatic taxi ranks will allow to significantly reduce the amount of fuss |

**Table 5-2:** A set of sentences covering significant letter sequences.

The original script was writing done in free space and no baseline correction technique has been applied.

In the following experiments, classification results were achieved on 606 test words randomly selected from the total data set of 3648 word images. The training set consisted of 3042 words, all of which were not in the test set.

Experimental results for binary classification (classification between every two classes) and triple classification (classification between three classes) are given in the following sections.

Note: In all the tables that follow, Nli, Nui and Nmi represent the training and test sets where N, u, l and m indicates the number of features, uppercase, lowercase and mixed case words respectively whilst i indicates the set number.

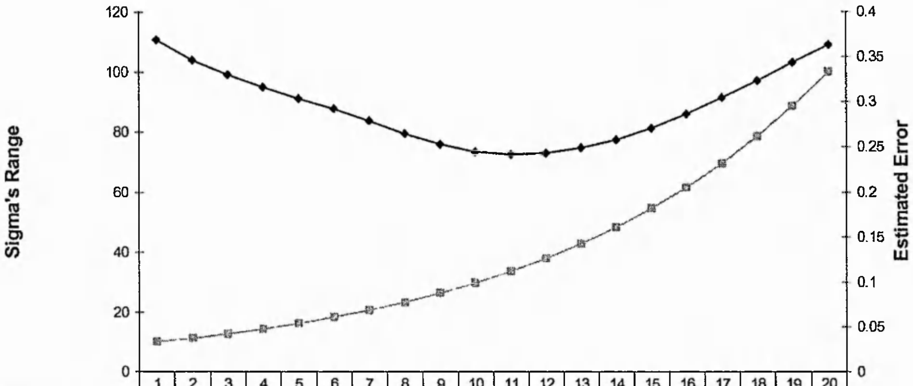
## 5.2 PNN using common $\sigma$

### 5.2.1 Binary classification

Tables 5-3 to 5-8 show the two class classification results obtained when using the non-linear (PNN) classification technique based on the selected values of common $\sigma$ applied on feature vectors of word images (see chapter 4 for more detail). In all of these tables the first column shows the samples that were used as the training data set whilst the second column shows the samples that were used as a test set. The third column shows the correct classification results obtained when using the non-linear (PNN) classification technique with a common $\sigma$ value. Rows 1 and 2 of the fourth column show the average correct classification results when the system was tested with seen and unseen data, respectively, and last row shows overall classification result for all data. A detailed analysis of all these results is presented in the following sections.

#### 5.2.1.1 Experimental results and analysis using 36 extracted features

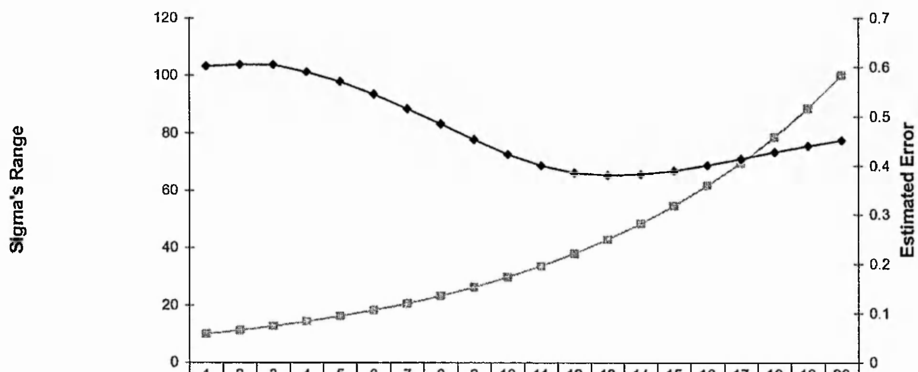
The results shown in figures 5-1 to 5-3 indicate that the best value of $\sigma$ lies within the 29 to 37 interval (calculated as 34.06559) for lower/upper case word images with an error rate of 0.24150. Note a logarithm transformation has been applied to compress the dynamic range of $\sigma$ (x axis). For mixed/lower case word images the best value of $\sigma$ lies within the 37 to 48 interval (calculated as 43.74458) with error rate of 0.38030 and for upper/mixed case word images the best value of $\sigma$ lies within the 29 to 37 interval (calculated as 32.31938) with an error rate of 0.23938. These are therefore the values used in the common $\sigma$ based PNN binary case classification experiments when using all 36 features.




| | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Sigma's Range | 10 | 11.3 | 12.7 | 14.4 | 16.2 | 18.3 | 20.7 | 23.4 | 26.4 | 29.8 | 33.6 | 37.9 | 42.8 | 48.3 | 54.6 | 61.6 | 69.5 | 78.5 | 88.6 | 100 |
| Error rate | 0.37 | 0.35 | 0.33 | 0.32 | 0.3 | 0.29 | 0.28 | 0.26 | 0.25 | 0.24 | 0.24 | 0.24 | 0.25 | 0.26 | 0.27 | 0.29 | 0.3 | 0.32 | 0.34 | 0.36 |

Number of Segmented Regions

**Figure 5-1:** Error estimation of common $\sigma$ for a classification of lower and upper case word images using all 36 extracted features ($\sigma$ =34.06559).




| | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Sigma's Rang | 10 | 11.3 | 12.7 | 14.4 | 16.2 | 18.3 | 20.7 | 23.4 | 26.4 | 29.8 | 33.6 | 37.9 | 42.8 | 48.3 | 54.6 | 61.6 | 69.5 | 78.5 | 88.6 | 100 |
| Estimated Error | 0.6 | 0.6 | 0.6 | 0.59 | 0.57 | 0.54 | 0.52 | 0.49 | 0.45 | 0.42 | 0.4 | 0.39 | 0.38 | 0.38 | 0.39 | 0.4 | 0.41 | 0.43 | 0.44 | 0.45 |

Number of segmented Sigma's Range

**Figure 5-2.** Error estimation of common $\sigma$ for a classification of lower and mixed case word images using all 36 extracted features ($\sigma$ =43.74458)

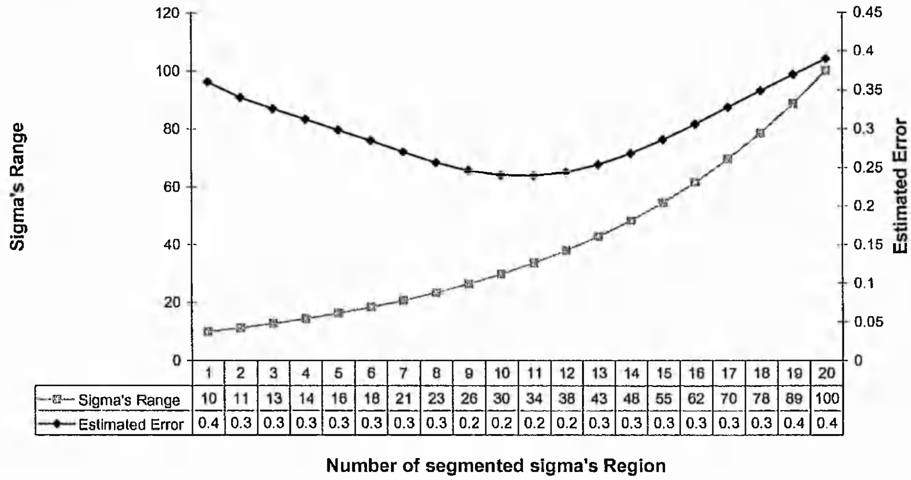



**Figure 5-3:** Error estimation of common $\sigma$ for a classification of upper and mixed case word images using all 36 extracted features ($\sigma$ =32.31938).

| Training sets | Test sets | %Correct Non-linear Classification (PNN) | %Correct Average |
|---|---|---|---|
| 36l2, 36u2 | 36u2 | 100.00% | 97.00% |
| 36l2, 36u2 | 36l2 | 94.00% | |
| 36l2, 36u2 | 36u1 | 96.00% | 83.50% |
| 36l2, 36u2 | 36l1 | 71.00% | |
| | | Overall | 90.25% |

**Table 5-3:** Classification result using all 36 features to discriminate between lower and upper case word images using common $\sigma$ ($\sigma$ =34.06559).

| Training sets | Test sets | %Correct Non-linear Classification (PNN) | %Correct Average |
|---|---|---|---|
| 36m2, 36u2 | 36u2 | 100.00% | 98.50% |
| 36m2, 36u2 | 36m2 | 97.00% | |
| 36m2, 36u2 | 36u1 | 93.00% | 84.50% |
| 36m2, 36u2 | 36m1 | 76.00% | |
| | | Overall | 91.50% |

**Table 5-4:** Classification result using all 36 features to discriminate between mixed and upper case word images using common $\sigma$ ($\sigma$ =32.31938).

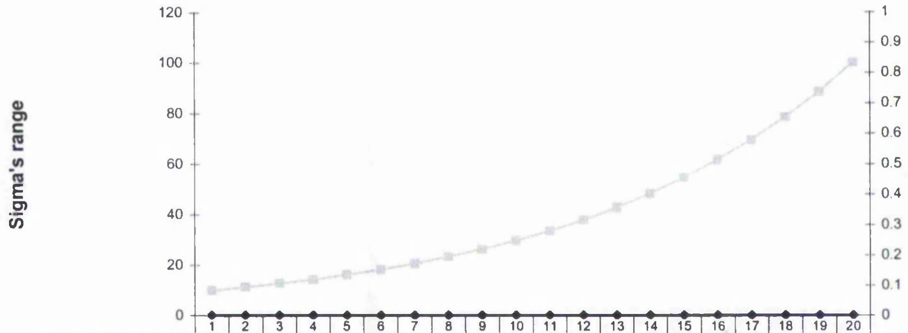
| Training sets | Test sets | %Correct Non-linear Classification (PNN) | %Correct Average |
|---|---|---|---|
| 36m2, 3612 | 36m2 | 99.00% | 97.00% |
| 36m2, 3612 | 3612 | 95.00% | |
| 36m2, 3612 | 36m1 | 81.00% | 73.00% |
| 36m2, 3612 | 3611 | 65.00% | |
| | | Overall | 85.00% |

**Table 5-5:** Classification result using all 36 features to discriminate between mixed and lower case word images using common $\sigma$ ($\sigma$ =43.74458).

Tables 5-3 to 5-5 show that the overall classification results are 90.25%, 91.50% and 85.00% when classifying lower/upper, mixed/upper and mixed/lower case word images respectively using all 36 extracted features. This can be broken down into 97.00%, 98.50% and 97.00% correct classification when the test set is the same as the training set and 83.50%, 84.50% and 73.00% correct classification when the test set is different to the training set.

### 5.2.1.2 Experimental results and analysis using the selected features

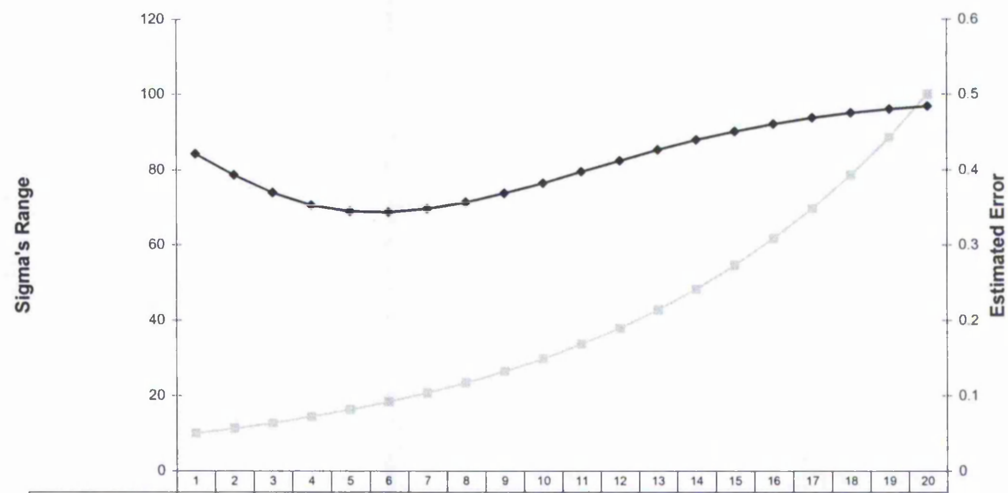
Section 4.4.1 (page 4-16) in chapter 4 shows the selected features. The results shown in figures 5-4 to 5-10 indicate that the best value of $\sigma$ lies within the 10 to 37 interval (calculated as 10.00) for lower/upper case word images with a zero error rate. For mixed/ lowercase word images the best value of $\sigma$ lies within the 16 to 20 interval (calculated as 18.32981) with error rate of 0.34296 and for upper/ mixed case word images the best value of $\sigma$ lies within the 10 to 14 interval (calculated as 10.67619) with error rate of 0.23000.




| | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Sigma's Range | 10 | 11.3 | 12.7 | 14.4 | 16.2 | 18.3 | 20.7 | 23.4 | 26.4 | 29.8 | 33.6 | 37.9 | 42.8 | 48.3 | 54.6 | 61.6 | 69.5 | 78.5 | 88.6 | 100 |
| Error Estimation | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

**Number of segmented sigma's Range**

**Figure 5-4:** Error estimation of common $\sigma$ for a classification of lower and upper case word images using 20 selected features ($\sigma$ =10.00000).




| | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Sigma's Ranger | 10 | 11.29 | 12.74 | 14.38 | 16.24 | 18.33 | 20.69 | 23.36 | 26.37 | 29.76 | 33.6 | 37.93 | 42.81 | 48.33 | 54.56 | 61.58 | 69.52 | 78.48 | 88.59 | 100 |
| Estimated Error | 0.421 | 0.393 | 0.369 | 0.353 | 0.344 | 0.343 | 0.348 | 0.356 | 0.368 | 0.382 | 0.397 | 0.412 | 0.427 | 0.44 | 0.451 | 0.461 | 0.469 | 0.475 | 0.48 | 0.485 |

**Number of segmented sigma's Range**

**Figure 5-5:** Error estimation of common $\sigma$ for a classification of lower and mixed case word images using 23 selected features ($\sigma$ =18.32981).

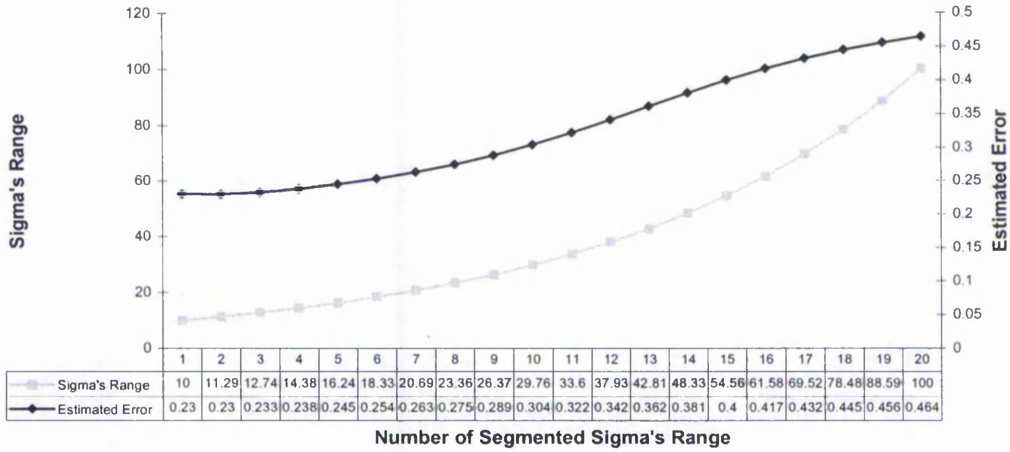



**Figure 5-6:** Error estimation of common $\sigma$ for a classification of upper and mixed case word images using 15 selected features ($\sigma$ =10.67619).

| Training sets | Test sets | %Correct Non-linear Classification (PNN) | %Correct Average |
|---|---|---|---|
| 20u2, 20l2 | 20u2 | 100.00% | 100.00% |
| 20u2, 20l2 | 20l2 | 100.00% | |
| 20u2, 20l2 | 20u1 | 100.00% | 100.00% |
| 20u2, 20l2 | 20l1 | 100.00% | |
| | | **Overall** | 100.00% |

**Table 5-6:** Classification result using 20 selected features to discriminate between lower and upper case word images using common $\sigma$ ($\sigma$ =10.00000).

| Training sets | Test sets | %Correct Non-linear Classification (PNN) | %Correct Average |
|---|---|---|---|
| 15m2, 15u2 | 15u2 | 96.00% | 91.50% |
| 15m2, 15u2 | 15m2 | 87.00% | |
| 15m2, 15u2 | 15u1 | 90.00% | 85.00% |
| 15m2, 15u2 | 15m1 | 80.00% | |
| | | **Overall** | 88.25% |

**Table 5-7:** Classification result using 15 selected features to discriminate between mixed and upper case word images using common $\sigma$ ($\sigma$ =10.67619).

| Training sets | Test sets | %Correct Non-linear Classification (PNN) | %Correct Average |
|---|---|---|---|
| 23m2, 2312 | 23m2 | 98% | 96.00% |
| 23m2, 2312 | 2312 | 94% | |
| 23m2, 2312 | 23m1 | 84% | 77.00% |
| 23m2, 2312 | 2311 | 70% | |
| | | Overall | 86.00% |

**Table 5-8:** Classification result using 23 selected features to discriminate between mixed and lower case word images using common $\sigma$ ($\sigma$ =18.32981).

Tables 5-6 to 5-8 show that the overall classification results are 100.00%, 88.25% and 86.50% when classifying between lower/upper, upper/mixed and mixed/lower case word images respectively. This can be broken down into 100.00%, 91.50% and 96.00% correct classification when the test set is the same as the training set and 100.00%, 85.00% and 77.00% correct classification when the test set is different to the training set.

### 5.2.1.3 Comparison between the selected and 36 extracted features

Figures 5-7 and 5-8 summarise the results shown in tables 5-3 to 5-8 (pages 5-6, 5-9 and 5-10). Figure 5-7 shows classification results when the training set is the same as the tests set and figure 5-8 shows classification results when the training set is different to the test set.

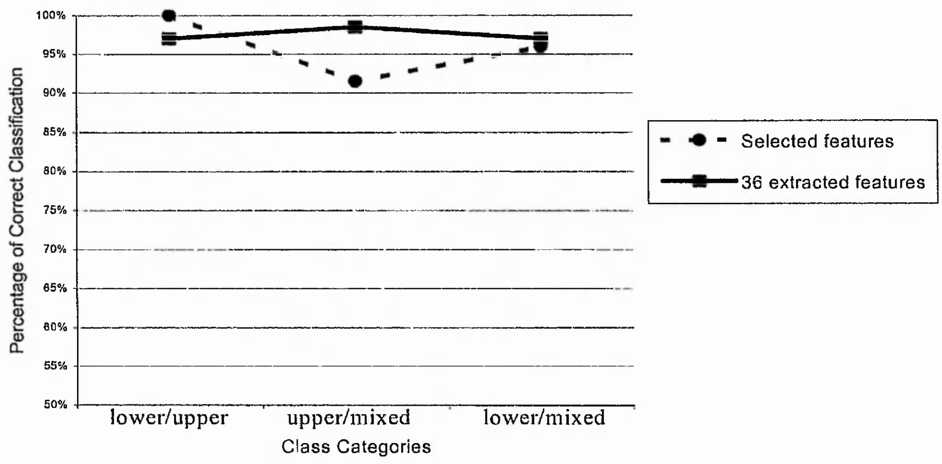



**Figure 5-7:** Comparison between using the selected features and all 36 features for seen data (common $\sigma$).

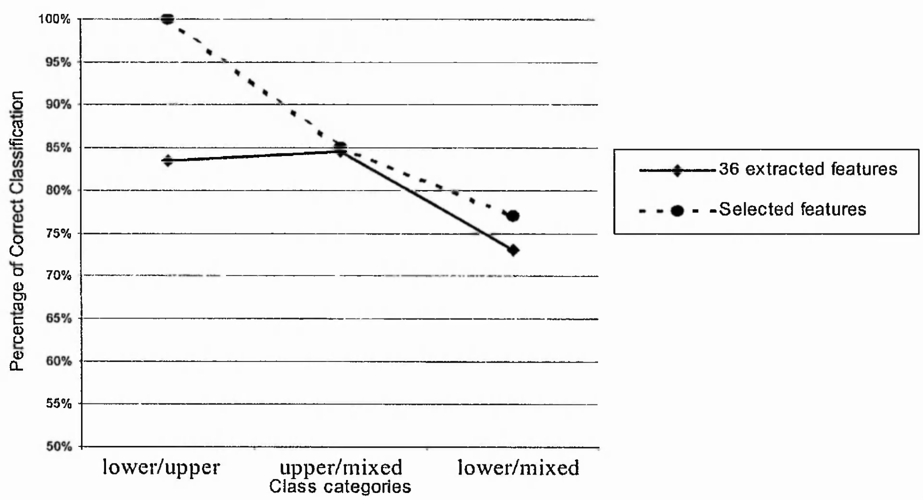



**Figure 5-8:** Comparison between using the selected features and all 36 features for unseen data (common $\sigma$).

Figure 5-7 shows that when training set is the same as the test set. An improvement of 3.00% and a decrease of 7.00% and 1.00% can be achieved in a classification between lower/upper, mixed/upper and mixed/lower case respectively when using the selected features compared to using the 36 using extracted features.

Figure 5-8 shows that when the test set is different to the training set an improvement of 16.500%, 0.50% and 4.00% was achieved when using the selected features for classifying between lower/upper, mixed/upper and mixed/lower case rather than using all 36 features.

### 5.2.2 Triple classification

Tables 5-9 (page 5-13) and 5-10 (page 5-15) give the results for the 3 class data sets. The first column shows the samples that were used as the training data set whilst the second column shows the samples that were used as a test set. The third column shows the correct classification results obtained when using the non-linear classification technique with common $\sigma$ value and either (i) using all 36 (table 5-9 page 5-13) or (ii) selected features (table 5-10 page 5-15). The fourth, fifth, sixth and seventh columns show the misclassification results in each category and the average of the classification results for seen and unseen data. The last row of column shows the overall classification result for all.

#### 5.2.2.1 Experimental results and analysis using 36 extracted features

For three class case classification the best value of common $\sigma$ is 38.97138. This lies within the 33 to 42 interval, with an error rate of 0.45755. The details are shown in figure 5-9.

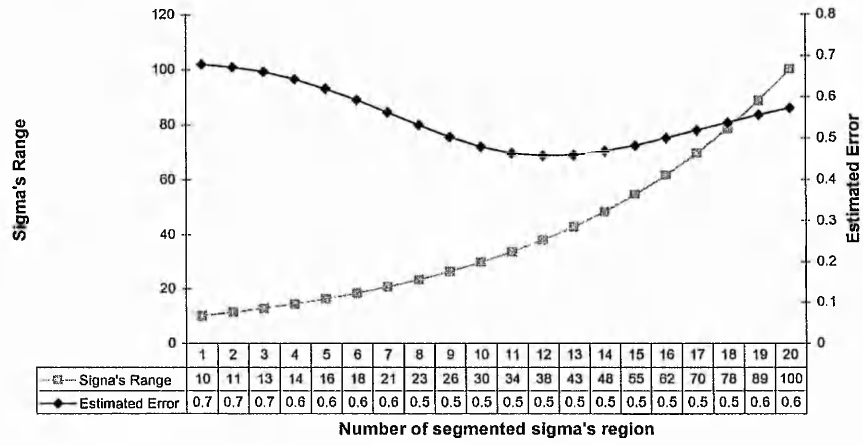



**Figure 5-9:** Error estimation based on common $\sigma$ for classification lower, upper and mixed case word images using 36 features ($\sigma$ =38.97138).

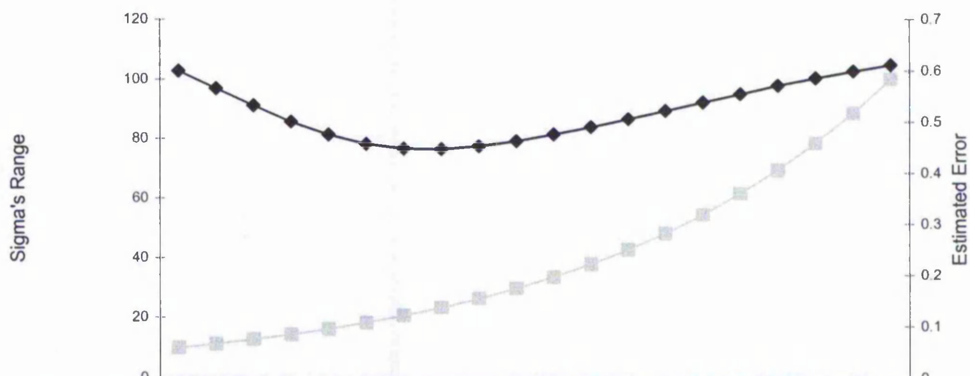
| Training set | Test sets | %Correct non-linear (PNN) | Misclassified words: As lower case | As upper case | As mixed case | %Correct Average |
|---|---|---|---|---|---|---|
| 36l2, 36u2, 36m2 | 36l2 | 89.00% | - | 10.00% | 1.00% | |
| 36l2,36u2,36m2 | 36u2 | 100.00% | 0 | - | 0 | 92.33% |
| 36l2,36u2, 36m2 | 36m2 | 88.00% | 0 | 12.00% | - | |
| 36l2,36u2, 36m2 | 36l1 | 54.00% | - | 24.00% | 22.00% | |
| 36l2,36u2, 36m2 | 36u1 | 93.00% | 3.00% | - | 4.00% | 68.00% |
| 36l2,36u2, 36m2 | 36m1 | 57.00% | 19.00% | 24.00% | - | |
| | | | | | Overall | 80.17% |

**Table 5-9:** Classification result using all 36 features to discriminate between mixed, lower and upper case word images using common $\sigma$ ($\sigma$ =38.97138).

The experimental results given in table 5-9 show that a classifier based on PNN using a common $\sigma$ can achieve 68.00% of correct case classification when the test set is different to the training set. The system can also be seen to achieve a 92.33% correct classification when test set is the same as the training set. This gives an overall 80.17% correct classification for all data.

### 5.2.2.2 Experimental results and analysis using selected features

For three class case classification, using the selected features, the best common value of $\sigma$ is 22.30066. This lies within the 16 to 23 interval with an error rate of 0.44569. The details are shown in the figure 5-10.




| | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Sigma's Range | 10 | 11.3 | 12.7 | 14.4 | 16.2 | 18.3 | 20.7 | 23.4 | 26.4 | 29.8 | 33.6 | 37.9 | 42.8 | 48.3 | 54.6 | 61.6 | 69.5 | 78.5 | 88.6 | 100 |
| Estimated Error | 0.6 | 0.57 | 0.53 | 0.5 | 0.47 | 0.46 | 0.45 | 0.45 | 0.45 | 0.46 | 0.48 | 0.49 | 0.51 | 0.52 | 0.54 | 0.55 | 0.57 | 0.59 | 0.6 | 0.61 |

Number of segmented sigma's range

**Figure 5-10:** Error estimation of common $\sigma$ for a triple classification using 25 selected features ($\sigma$ =22.30060).

| Training sets | Test sets | %Correct non-linear (PNN) | Misclassified words | | | %Correct Average |
|---|---|---|---|---|---|---|
| | | | As lower case | As upper case | As mixedcase | |
| 2512, 25u2, 25m2 | 2512 | 86.00% | - | 12.00% | 2.00% | |
| 2512, 25u2, 25m2 | 25u2 | 98.00% | 2.00% | - | - | 91.33% |
| 2512, 25u2, 25m2 | 25m2 | 90.00% | 2.00% | 8.00% | - | |
| 2512, 25u2, 25m2 | 2511 | 52.00% | - | 26.00% | 22.00% | |
| 2512, 25u2, 25m2 | 25u1 | 90.00% | 5.00% | - | 5.00% | 66.67% |
| 2512, 25u2, 25m2 | 25m1 | 58.00% | 24.00% | 18.00% | - | |
| | | | | | Overall | 79.00% |

**Table 5-10:** Classification result using selected features to discriminate between lower, upper and mixed case word images using common $\sigma$ ($\sigma$ =22.30066).

Table 5-10 gives the results for the three class data sets using the 25 selected features. Overall, these experimental results show that a classifier based on PNN using a common $\sigma$ can achieve a 66.67% correct case classification when the test set is different to the training set. The system can also be seen to achieve a 91.33% correct classification when the test set is the same as the training set. This gives an overall 79.00% correct classification for all data.

### 5.2.2.3 Comparison between using the selected and 36 extracted features

Figures 5-11 and 5-12 summarise the results from tables 5-9 and 5-10 (pages 5-113and 5-15). Figure 5-11 shows the classification results when the test set is the same as the training set and figure 5-12 shows the classification result when the test set is different to the training set. In both experiments the results using all 36 features are better than results obtained when using the selected features.

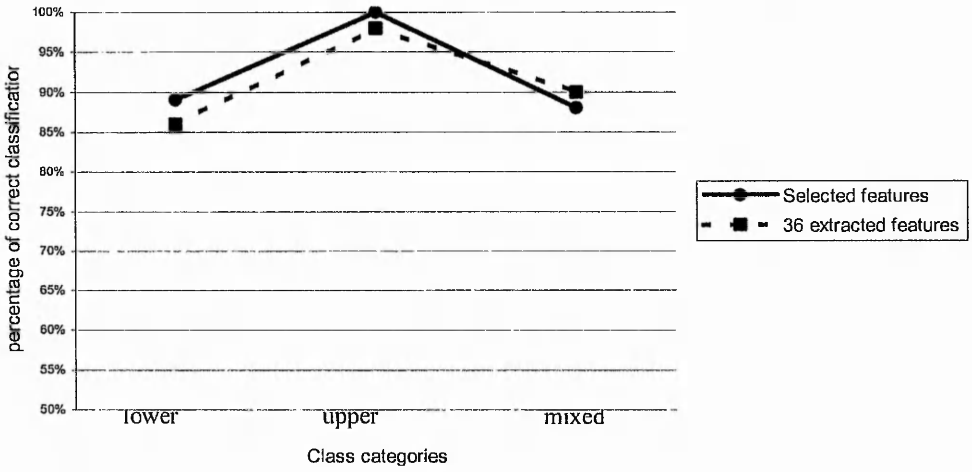



**Figure 5-11:** Comparison between the selected and all 36 features with common $\sigma$ value for seen data.

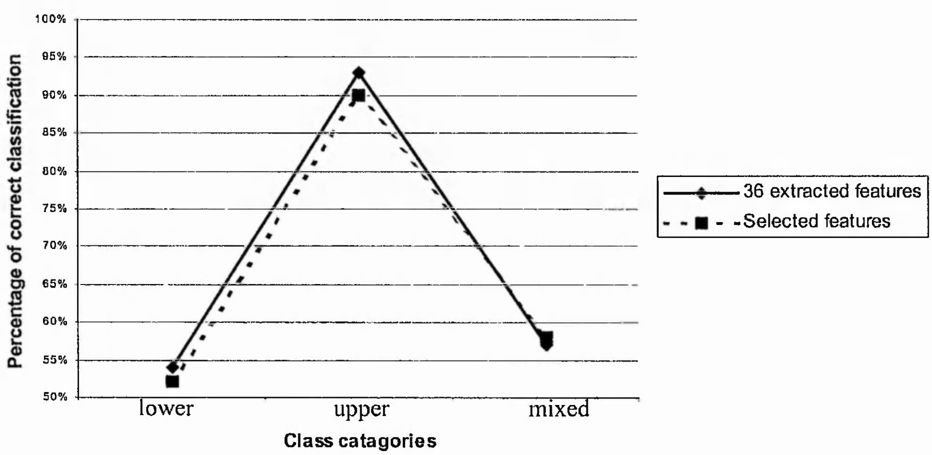



**Figure 5-12:** Comparison between the selected features and all 36 features with a common $\sigma$ value for unseen data.

Figure 5-11 shows that an improvement of 3.00%, 2.00% and decrease 2.00% can be achieved for lower, upper and mixed case words classification for seen data and an improvement of 2.00% and 3.00% for lower and upper case classification and a decrease of 1.00% for mixed case word can be achieved for classification with 36 features in comparison to using the 25 selected features.

## 5.3 PNN using different $\sigma_i$

### 5.3.1 Binary classification

Tables 5-12 to 5-14 (page 5-19) and 5-16 to 5-18 (page 5-21) show the classification results obtained when using a non-linear classification (PNN) technique based on different values of $\sigma_i$ $(i = 1, 2, \cdots, 36)$. The first column in these tables shows the samples that were used as the training data set whilst the second column shows the samples that were used as a test set. The third column shows the correct classification results obtained when using a non-linear classification (PNN) technique with different $\sigma_i$ using either (i) all 36 (tables 5-12 to 5-14) and (ii) the selected features (tables 5-16 to 5-18). The fourth column shows the average of the classification result for seen and unseen data. The last row shows average classification result for all data.

### 5.3.1.1 Experimental results and analysis using 36 extracted features

Table 5-11 shows the best values of different $\sigma_i$ for triple classification using 36 features.

| Lower/upper $\sigma_i$ in lower class | Lower/upper $\sigma_i$ in mixed class | Lower/mixed $\sigma_i$ in lower class | Lower/mixed $\sigma_i$ in mixed class | Upper/mixed $\sigma_i$ in upper class | Upper/mixed $\sigma_i$ in mixed class |
|---|---|---|---|---|---|
| 31.66910 | 36.50039 | 40.30089 | 46.95132 | 33.11365 | 30.87433 |
| 33.21365 | 32.91449 | 42.10445 | 40.45395 | 33.09760 | 27.56992 |
| 35.98528 | 25.05902 | 38.40030 | 26.49945 | 20.65024 | 31.43430 |
| 34.22086 | 28.88814 | 43.28656 | 26.26551 | 21.56173 | 36.52140 |
| 25.6966 | 35.00917 | 47.94237 | 25.30436 | 11.36517 | 38.51632 |
| 31.72962 | 36.83665 | 40.01519 | 46.77799 | 32.07831 | 31.85892 |
| 35.71564 | 3.563994 | 50.06942 | 39.56078 | 32.81210 | 33.21455 |
| 31.19477 | 3.684172 | 45.96327 | 44.45081 | 28.07341 | 33.86134 |
| 33.09749 | 33.70813 | 43.48732 | 44.99848 | 29.79570 | 32.87055 |
| 31.71058 | 37.15968 | 40.47494 | 47.36407 | 33.12519 | 31.69331 |
| 31.59668 | 37.23329 | 40.33389 | 47.62987 | 33.07537 | 31.77327 |
| 31.91357 | 35.93845 | 37.00052 | 44.24160 | 32.25076 | 32.64724 |
| 30.21646 | 36.81034 | 39.20706 | 48.36870 | 3.11558 | 31.82481 |
| 30.52064 | 36.49550 | 36.30819 | 43.54169 | 31.55467 | 32.17488 |
| 36.21306 | 39.14792 | 58.59066 | 64.45274 | 48.24837 | 4.582945 |
| 41.23781 | 47.21730 | 68.00391 | 75.52854 | 35.73442 | 43.59700 |
| 40.45418 | 51.41041 | 74.76022 | 67.82258 | 40.51177 | 39.60755 |
| 50.29690 | 39.24159 | 62.94516 | 67.00644 | 48.09203 | 53.84457 |
| 47.82275 | 46.61158 | 49.90040 | 50.08594 | 45.09720 | 55.91957 |
| 54.64093 | 59.14819 | 86.64230 | 66.81278 | 63.44285 | 63.57912 |
| 46.78131 | 49.35349 | 65.84567 | 55.52525 | 50.22523 | 51.68060 |
| 31.39516 | 46.34556 | 73.66818 | 56.20340 | 48.12154 | 54.39196 |
| 45.505 | 58.66783 | 69.91064 | 71.23247 | 57.78916 | 67.22363 |
| 32.48761 | 27.93953 | 39.21604 | 43.95092 | 36.62340 | 23.18794 |
| 31.67121 | 37.23184 | 40.2831 | 47.64449 | 33.14543 | 31.73185 |
| 40.36785 | 24.66710 | 29.21175 | 3.097868 | 35.35132 | 16.94857 |
| 30.62371 | 36.69095 | 25.04309 | 44.47799 | 28.44528 | 30.24667 |
| 25.17725 | 18.73598 | 47.15635 | 28.58840 | 16.647491 | 12.01343 |
| 34.59646 | 34.31782 | 44.35184 | 43.48665 | 32.24041 | 32.72930 |
| 16.91427 | 79.99161 | 23.79367 | 25.7205 | 13.88777 | 12.75120 |
| 34.6101 | 22.43994 | 34.51756 | 17.45586 | 20.69082 | 38.47681 |
| 30.03060 | 23.58392 | 10.52882 | 10.07357 | 11.98927 | 14.24093 |
| 31.70338 | 33.31331 | 37.26071 | 31.86102 | 36.66826 | 22.76129 |
| 37.26955 | 23.61908 | 15.27425 | 31.31350 | 50.90226 | 10.68675 |
| 38.39426 | 29.05656 | 44.51663 | 57.74372 | 31.92841 | 31.51337 |
| 40.54235 | 31.06225 | 30.23730 | 48.70842 | 33.77714 | 16.03300 |

**Table 5-11:** Thirty-six different $\sigma_i$ for each lower/upper, mixed/lower and lower/mixed case classes using the 36 extracted features.

Table 5-11 shows the best values of different $\sigma_i$ for each lower/upper, upper/mixed and lower/mixed case classes obtained as explained in chapter 4. The error rate for upper/lower, upper/mixed and lower/mixed class is 0.24356, 0.16949 and 0.29464.

| Training sets | Test sets | %Correct Non-linear Classification (PNN) | %Correct Average |
|---|---|---|---|
| 36l2, 36u2 | 36u2 | 100.00% | 97.50% |
| 36l2, 36u2 | 36l2 | 95.00% | |
| 36l2, 36u2 | 36u1 | 91.00% | 85.00% |
| 36l2, 36u2 | 36l1 | 79.00% | |
| | | Overall | 91.25% |

**Table 5-12:** Classification result using all 36 features to discriminate between lower and upper case words using different $\sigma_i$ .

| Training sets | Test sets | %Correct Non-linear Classification (PNN) | %Correct Average |
|---|---|---|---|
| 36m2, 36u2 | 36u2 | 100.00% | 99.50% |
| 36m2, 36u2 | 36m2 | 99.00% | |
| 36m2, 36u2 | 36u1 | 91.00% | 88.00% |
| 36m2, 36u2 | 36m1 | 85.00% | |
| | | Overall | 93.75% |

**Table 5-13:** Classification result using all 36 features to discriminate between mixed and upper case words using different $\sigma_i$ .

| Training sets | Test sets | %Correct Non-linear Classification (PNN) | %Correct Average |
|---|---|---|---|
| 36m2, 36l2 | 36m2 | 99.00% | 97.50% |
| 36m2, 36l2 | 36l2 | 96.00% | |
| 36m2, 36l2 | 36m1 | 83.00% | 81.00% |
| 36m2, 36l2 | 36l1 | 79.00% | |
| | | Overall | 89.25% |

**Table 5-14:** Classification result using all 36 features to discriminate between mixed and lower case word using different $\sigma_i$ .

Tables 5-12 to 5-14 show that the overall classification results are 91.25%, 93.75% and 89.25% correct classification when classifying between lower/upper, mixed/upper and mixed/lower case word images respectively. The system can also achieve 97.50%, 99.50% and 97.50% correct classification when the test set is the same as the training set and 85.00%, 88.00% and 81.00% correct classification when the test set is different from training set.

### 5.3.1.2 Experimental results and analysis using selected features

Table 5-15 shows the best values of different $\sigma_i$ for each lower/upper, mixed/lower and lower/mixed case classes obtained as explained in chapter 4 section 4.3 using different number of selected features for each binary classification. The error rate for upper/lower, upper/mixed and lower/mixed class is 0.0, 0.16949 and 0.29464 respectively.

| Lower/upper $\sigma_i$ in lower class | Lower/upper $\sigma_i$ in upper class | Lower/mixed $\sigma_i$ in lower class | Lower/mixed $\sigma_i$ in mixed class | Upper/mixed $\sigma_i$ in upper class | Upper/mixed $\sigma_i$ in mixed class |
|---|---|---|---|---|---|
| 8.85866 | 8.85866 | 18.06529 | 18.32134 | 11.61462 | 1.74298 |
| 8.85866 | 8.8586 | 18.45531 | 15.46795 | 6.58615 | 6.42734 |
| 8.85866 | 8.85866 | 19.64883 | 13.83216 | 8.11549 | 12.96917 |
| 8.85866 | 8.85866 | 22.28540 | 14.91802 | 11.17791 | 11.86621 |
| 8.85866 | 8.85866 | 26.95118 | 15.68670 | 12.03900 | 9.89083 |
| 8.85866 | 8.85866 | 25.70221 | 19.71557 | 15.00805 | 10.80199 |
| 8.85866 | 8.85866 | 20.77041 | 16.56491 | 8.70104 | 18.99999 |
| 8.85866 | 8.85866 | 22.66996 | 21.56909 | 18.42362 | 11.55111 |
| 8.85866 | 8.85866 | 18.34385 | 18.38123 | 11.54051 | 11.09565 |
| 8.85866 | 8.85866 | 18.17623 | 18.55501 | 10.35323 | 11.91820 |
| 8.85866 | 8.85866 | 12.22191 | 21.35154 | 7.44281 | 7.26623 |
| 8.85866 | 8.85866 | 14.19849 | 19.30047 | 5.36526 | 4.34294 |
| 8.85866 | 8.85866 | 16.65969 | 17.39400 | 11.43165 | 10.15984 |
| 8.85866 | 8.85866 | 18.14806 | 18.47655 | 1.27919 | 8.89669 |
| 8.85866 | 8.85866 | 3.396840 | 17.90655 | 17.20104 | 12.79321 |
| 8.85866 | 8.85866 | 18.58524 | 17.89910 | - | - |
| 8.85866 | 8.85866 | 23.98486 | 16.15866 | - | - |
| 8.85866 | 8.85866 | 23.26921 | 13.92327 | - | - |
| 8.85866 | 8.85866 | 11.16574 | 13.21324 | - | - |
| 8.85866 | 8.85866 | 13.05373 | 14.09981 | - | - |
| - | - | 18.25853 | 23.10075 | - | - |
| - | - | 24.27331 | 22.58113 | - | - |
| - | - | 23.47320 | 28.11260 | - | - |

**Table 5-15:** Different $\sigma_i$ value for each lower/upper, mixed/lower and lower/mixed case classes using selected features.

| Training sets | Test sets | %Correct Non-linear Classification (PNN) | %Correct Average |
|---|---|---|---|
| 20u2, 2012 | 20u2 | 97.00% | 98.00% |
| 20u2, 2012 | 2012 | 99.00% | |
| 20u2, 2012 | 20u1 | 89.00% | 84.50% |
| 20u2, 2012 | 2011 | 80.00% | |
| | | Overall | 91.25% |

**Table 5-16:** Classification result using the 20 selected features to discriminate between upper and lower case word with different $\sigma_i$.

| Training sets | Test sets | %Correct Non-linear Classification (PNN) | %Correct Average |
|---|---|---|---|
| 15m2, 15u2 | 15u2 | 96.00% | 94.00% |
| 15m2, 15u2 | 15m2 | 92.00% | |
| 15m2, 15u2 | 15u1 | 90.00% | 87.00% |
| 15m2, 15u2 | 15m1 | 84.00% | |
| | | Overall | 90.50% |

**Table 5-17:** Classification result using the 15 selected features to discriminate between mixed and upper case word with different $\sigma_i$.

| Training sets | Test sets | %Correct Non-linear Classification (PNN) | %Correct Average |
|---|---|---|---|
| 23m2, 2312 | 23m2 | 98.00% | 96.00% |
| 23m2, 2312 | 23612 | 94.00% | |
| 23m2, 2312 | 23m1 | 84.00% | 77.50% |
| 23m2, 2312 | 2311 | 71.00% | |
| | | Overall | 86.75% |

**Table 5-18:** Classification result using the 23 selected features to discriminate between mixed and lower case word with different $\sigma_i$.

Tables 5-16 to 5-18 show that the overall classification results are 91.25%, 90.50% and 86.75% correct classification when classifying lower/upper, mixed/upper and mixed/lower case word images respectively using selected features. The system also achieved 98.00%, 94.00% and 96.00% correct classification when the test set is the same as the training set and 84.50%, 87.00% and 77.50% correct classification when the test set is different from training set.

### 5.3.1.3 Comparison between using the selected and 36 extracted features

Figures 5-13 and 5-14 summarise the results of tables 5-12 to 5-14 (page 5-19) and 5-16 to 5-18 (pages 5-21). Figure 5-13 shows the classification result when the test set is the same as training set and figure 5-14 shows the classification result when the training set is different to the test set.

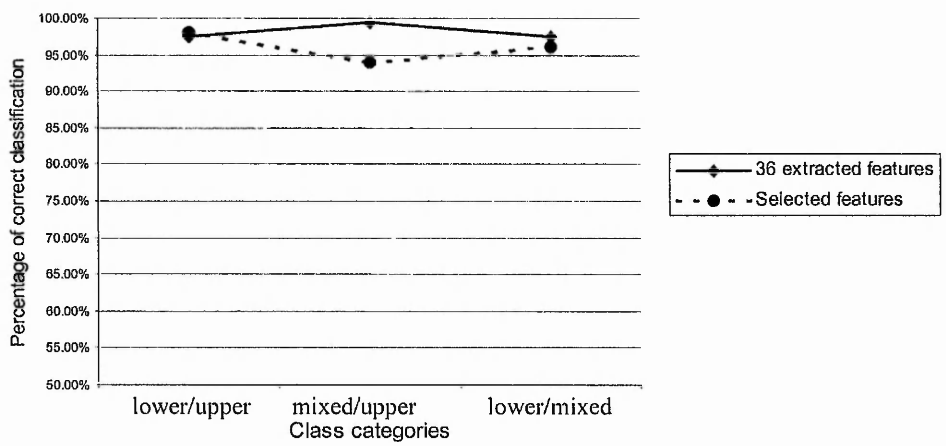



**Figure 5-13:** Comparison between the selected features and all 36 features using different $\sigma_i$ for seen data.

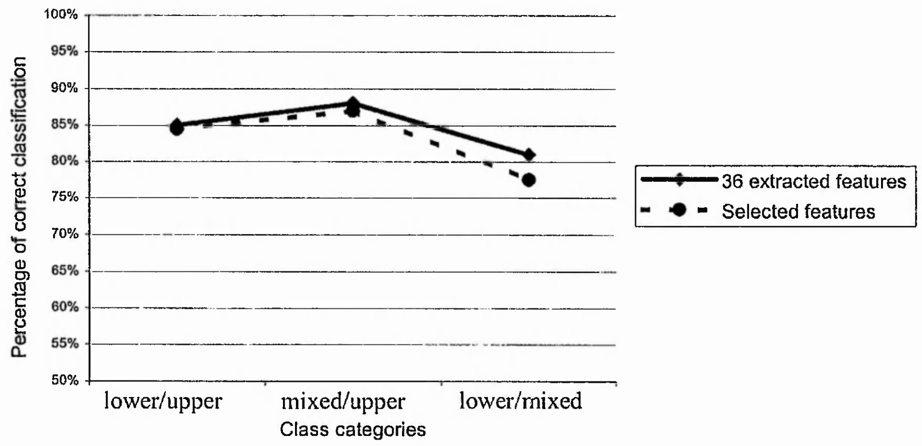



**Figure 5-14:** Comparison between the selected features and all 36 features using different $\sigma_i$ for unseen data.

These figures show that an improvement of 0.50%, 1.00% and 3.50% can be achieved in the classification of lower/upper, mixed/upper and mixed/lower when using 36 features in comparison to using the selected features while training set is different to the test set. Figures 5-13 and 5-14 show that the 36 features give better classification results than the selected features when using different $\sigma_i$ values.

### 5.3.2 Triple Classification

Table 5-20 and 5-22 shows the experimental results obtained when using all 36 or the 25 selected features for three class (upper, lower, mixed) case classification. The first column shows the samples that were used as the training data set whilst the second column shows the samples that were used as a test set. The third column in table 5-8 shows the correct classification results obtained when using the non-linear (PNN) classification technique using different $\sigma_i$ with 36 extracted features whilst column three shows the similar result when using the 25 selected features. Columns four, five and six then show the misclassification results in each category and the average of classification result for seen and unseen data. Finally the last row shows the overall classification result for all data.

### 5.3.2.1 Experimental results and analysis using 36 extracted features

Table 5-19 shows the best values of different $\sigma_i$ obtained for each lower, upper and mixed case class. The error rate for this experiment is 0.34282.

| Lower/upper/mixed $\sigma_i$ in lower class | Lower/upper/mixed $\sigma_i$ in upper class | Lower/upper/mixed $\sigma_i$ in mixed class |
|---|---|---|
| 34.60663 | 40.78809 | 40.15208 |
| 38.80092 | 30.51066 | 37.62117 |
| 28.4876 | 29.51596 | 23.55117 |
| 32.27446 | 35.90650 | 25.50454 |
| 36.43756 | 39.35751 | 13.68393 |
| 34.43485 | 41.80369 | 39.63732 |
| 46.10676 | 42.32424 | 33.89650 |
| 37.33797 | 43.55043 | 35.99928 |
| 40.94736 | 39.45335 | 36.77933 |
| 34.8720 | 42.02187 | 40.10461 |
| 34.65855 | 42.11689 | 40.24011 |
| 31.87318 | 37.78838 | 39.46066 |
| 32.97020 | 41.09790 | 40.34337 |
| 31.51717 | 40.99078 | 37.88076 |
| 48.44657 | 53.90671 | 45.34138 |
| 41.90354 | 60.49810 | 52.45157 |
| 60.23403 | 64.90404 | 55.47124 |
| 59.44048 | 48.28614 | 60.84754 |
| 47.24551 | 61.15829 | 49.31216 |
| 81.38790 | 77.06012 | 67.30111 |
| 60.49656 | 49.86284 | 50.00692 |
| 57.85636 | 52.43365 | 50.80648 |
| 73.18588 | 74.89508 | 61.25866 |
| 39.05878 | 25.48280 | 40.99422 |
| 34.68910 | 42.09919 | 40.27009 |
| 28.21441 | 20.89339 | 28.42728 |
| 24.98735 | 40.52817 | 36.32665 |
| 34.99968 | 13.27194 | 19.12642 |
| 41.03071 | 38.45952 | 36.96268 |
| 22.9680 | 8.95491 | 21.78347 |
| 45.44620 | 22.90310 | 19.82204 |
| 9.97458 | 13.05376 | 9.23380 |
| 38.38662 | 31.93748 | 34.40062 |
| 16.27195 | 13.75755 | 43.89245 |
| 34.79694 | 30.89873 | 51.49939 |
| 25.03110 | 17.87146 | 51.79355 |

**Table 5-19:** Thirty-six different $\sigma_i$ for each lower, upper and mixed case class using 36 extracted features.

| Training sets | Test sets | %Correct non-linear (PNN) | Misclassified words: As lower case | Misclassified words: As upper case | Misclassified words: As mixedcase | %Correct Average |
|---|---|---|---|---|---|---|
| 3612, 36u2, 36m2 | 3612 | 92.00% | - | 6.00% | 2.00% | 96.33% |
| 3612, 36u2, 36m2 | 36u2 | 100.00% | 0 | - | 0 | |
| 3612, 36u2, 36m2 | 36m2 | 97.00% | 1.00% | 2.00% | - | |
| 3612, 36u2, 36m2 | 3611 | 62.00% | - | 19.00% | 19.00% | 73.00% |
| 3612, 36u2, 36m2 | 36u1 | 87.00% | 7.00% | - | 6.00% | |
| 3612, 36u2, 36m2 | 36m1 | 70.00% | 28.00% | 2.00% | - | |
| | | | | | Overall | 84.67% |

**Table 5-20:** Classification results using all 36 features to discriminate between lower, upper and mixed case word using different $\sigma_i$.

These results show that the PNN classifier using different $\sigma_i$ achieves a 73.00% correct case classification when the test set is different to the training set and a 96.33% correct classification when the test set is the same as the training set. This gives an overall 84.67% correct classification result.

### 5.3.2.2 Experimental result and analysis using the selected features

Table 5-21 shows the best values of different $\sigma_i$ obtained for each lower, upper and mixed case class. The error rate for this experiment is 0.39681.

| Lower/upper/mixed $\sigma_i$ in lower class | Lower/upper/mixed $\sigma_i$ in upper class | Lower/upper/mixed $\sigma_i$ in mixed class |
|---|---|---|
| 21.72291 | 22.90424 | 23.82147 |
| 29.27063 | 15.52450 | 23.66054 |
| 29.82526 | 17.96224 | 21.31411 |
| 29.79063 | 28.50866 | 17.62378 |
| 23.03790 | 23.84364 | 22.87610 |
| 28.5845 | 28.53945 | 19.71792 |
| 25.34981 | 27.64416 | 20.18368 |
| 28.77008 | 24.06526 | 22.03334 |
| 22.74081 | 24.18985 | 23.44759 |
| 22.56603 | 24.30937 | 23.63730 |
| 18.99557 | 22.35720 | 21.58160 |
| 12.46167 | 22.45633 | 21.31705 |
| 14.10072 | 21.32987 | 20.24646 |
| 35.0381 | 44.62663 | 36.18569 |
| 33.48777 | 46.14005 | 31.75675 |
| 34.95456 | 34.36652 | 30.07377 |
| 16.56870 | 11.70558 | 16.32112 |
| 22.49589 | 24.27092 | 23.72664 |
| 20.72942 | 17.85299 | 22.68342 |
| 1.50826 | 21.90581 | 14.76931 |
| 26.68006 | 14.23823 | 16.09809 |
| 29.53675 | 17.62612 | 22.23733 |
| 25.11118 | 14.05060 | 19.30088 |
| 27.04151 | 17.22248 | 37.02396 |
| 25.53826 | 12.80108 | 26.65230 |

**Table 5-21:** Twenty-five different $\sigma_i$ for lower, upper and mixed case class using the selected features.

| | | | Misclassified words | | | |
|---|---|---|---|---|---|---|
| **Training sets** | **Test sets** | **%Correct non-linear (PNN)** | **As lower case** | **As upper case** | **As mixedcase** | **%Correct Average** |
| **25l2, 25u2, 25m2** | **25l2** | **93.00%** | **-** | **6.00%** | **1.00%** | |
| **25l2, 25u2, 25m2** | **25u2** | **99.00%** | **1.00%** | **-** | **0** | **92.67%** |
| **25l2, 25u2, 25m2** | **25m2** | **86.00%** | **9.00%** | **5.00%** | **-** | |
| **25l2, 25u2, 25m2** | **25l1** | **62.00%** | **-** | **22.00%** | **16.00%** | |
| **25l2, 25u2, 25m2** | **25u1** | **84.00%** | **13.00%** | **-** | **3.00%** | **71.67%** |
| **25l2, 25u2, 25m2** | **25m1** | **69.00%** | **22.00%** | **9.00%** | **-** | |
| | | | | | **Overall** | **82.17%** |

**Table 5-22:** Three case classification results using the 25 selected features to discriminate between lower, upper and mixed case word with different $\sigma_i$ .

Table 5-22 shows the experimental results obtained using 25 selected features for three class (upper, lower and mixed) classification. The system achieved 92.67% correct classification when the test set is the same as training set and 71.67% correct classification when the test set is different to the training set. This gives an overall 82.17% correct classification when using different $\sigma_i$ with the 25 selected features using different $\sigma_i$.

### 5.3.2.3 Comparison between using the selected and 36 extracted features

Figures 5-15 and 5-16 show that in a classification between upper/mixed and lower/mixed for seen data the classification result using 36 features is better than for lower/upper case words. In these figures x-axis line indicate the lower/upper (1), upper/mixed (2) and lower/mixed (3) case classes respectively. And for unseen data using the 36 extracted features gives better classification result than using 25 selected features.

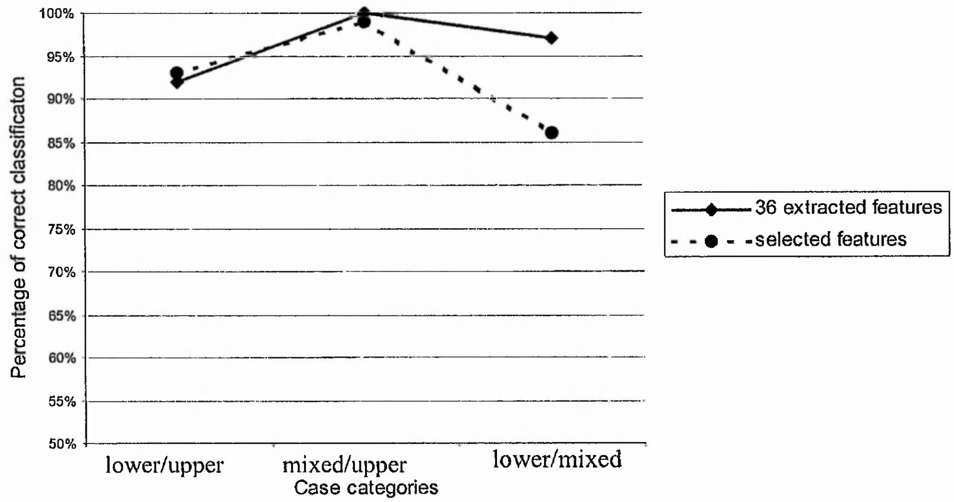



**Figure 5-15:** Comparison between the selected features and all 36 features with different $\sigma_i$ for seen data.

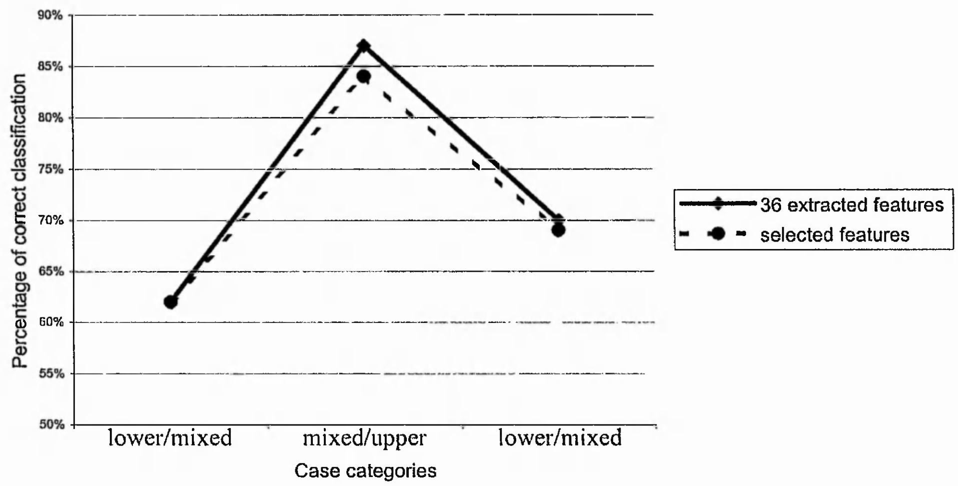



**Figure 5-16:** Comparison between the selected features and all 36 features with different $\sigma_i$ for unseen data.

Figure 5-15 compares the experimental results obtained when using selected features to those obtained when using the 36 features when training set is the same as the test set. This comparison shows that a decrease of 1.00% and an improvement of 1.00% and 11.00% can be achieved in a classification between lower, upper and mixed case word images when using 36 features rather than the selected features for case classification on seen data respectively. An improvement of 3.00% and 1.00% can also be achieved in classification between upper and mixed case words respectively for case classification of unseen data.

The results for test sets 36u1 and 25u1 in tables 5-20 and 5-22 also show that when the training set is different to the test set the correct classification rate for upper case word images using the 25 selected features is the same as when using the 36 features. This indicates that the rest of the features are only needed for classification between the lower and mixed case word images.

## 5.4 Comparison between the selected and 36 extracted features in Triple Classification using PNN method

The following table summarises the triple classification results achieved in the previous sections.

| Training set is different with the test set | Lowercase | | Uppercase | | Mixedcase | | Overall | |
|---|---|---|---|---|---|---|---|---|
| | Dif $\sigma_i$ | Com $\sigma$ | Dif $\sigma_i$ | Com $\sigma$ | Dif $\sigma_i$ | Com $\sigma$ | Dif $\sigma_i$ | Com $\sigma$ |
| Selected features | 62.00% | **54.00%** | 87.00% | **93.00%** | 70.00% | **57.00%** | 73.00% | **68.00%** |
| 36 extracted features | 62.00% | **52.00%** | 84.00% | **90.00%** | 69.00% | **58.00%** | 71.67% | **66.67%** |

**Table 5-23:** Comparison between classification results using different $\sigma_i$ and common $\sigma$ with selected features and 36 extracted features when the training set is different to the test set.

| Training set is the same as test set | Lowercase | | Uppercase | | Mixedcase | | Overall | |
|---|---|---|---|---|---|---|---|---|
| | Dif $\sigma_i$ | Com $\sigma$ | Dif $\sigma_i$ | Com $\sigma$ | Dif $\sigma_i$ | Com $\sigma$ | Dif $\sigma_i$ | Com $\sigma$ |
| Selected features | 92.00% | **89.00%** | 100.00% | **100.00%** | 97.00% | **88.00%** | 96.33% | **92.33%** |
| 36 extracted features | 93.00% | **86.00%** | 99.00% | **98.00%** | 86.00% | **90.00%** | 92.66% | **91.33%** |

**Table 5-24:** Comparison between classification results using different $\sigma_i$ and common $\sigma$ with selected features and 36 extracted features when the training set is the same as the test set.

### 5.4.1 36 Extracted features

The experimental result given in tables 5-23 and 5-24 show that in triple classification an improvement of 10.00% and 11.00% is achieved in the classification of lower and mixed case words using different $\sigma_i$ in comparison to using the common $\sigma$ when the training set is different to the test set. However, a decrease of 6.00% is achieved in the classification of upper case words. These tables also show that an improvement of 7.00% and 1.00% is achieved for lower and upper case words in triple classification when using different $\sigma_i$ comparison to the common $\sigma$ when the training set is the same as the test set. A decrease of 4.00% is also achieved for mixed case words.

Overall then, these tables show that using different $\sigma_i$ values with all 36 features can help the classifier to discriminante between lower and mixed case word images better than when using common $\sigma$ with all 36 features.

### 5.4.2 Selected features

The experimental results shown in tables 5-23 and 5-24 show that in triple classification an improvement of 8.00% and 13.00% is achieved in the classification of lower and mixed case words when the training set is different to the test set using different $\sigma_i$ in comparison to using the common $\sigma$ values. However, a decrease of 6.00% is achieved in classification of upper case words in triple classification. These tables also show that an improvement of 3.00% and 9.00% is achieved when using different $\sigma_i$ compared to using the common $\sigma$ when the training set is the same as the test set.

## 5.5 Multiple Linear Classification (MDA)

Tables 5-25 to 5-27 show the experimental results obtained using all 36 extracted features to classify between upper/lower, upper/mixed and lower/mixed case word images when using the multiple linear discriminant analysis technique. The first column shows the samples that were used as the training data set while the second column shows the samples that were used as a test set. The third column shows the correct classification results.

| Training sets | Test sets | %Correct Linear Classification (MDA) | %Correct Average |
|---|---|---|---|
| 36l2, 36u2 | 36u2 | 93.00% | 92.50% |
| 36l2, 36u2 | 36l2 | 82.00% | |
| 36l2, 36u2 | 36u1 | 90.00% | 84.00% |
| 36l2, 36u2 | 36l1 | 78.00% | |
| | | Overall | 88.25% |

**Table 5-25:** Classification result using all 36 features to discriminate between lower and upper case word images with the MDA technique.

| Training sets | Test sets | %Correct Linear Classification (MDA) | %Correct Average |
|---|---|---|---|
| 36m2, 36u2 | 36u2 | 93.00% | 88.50% |
| 36m2, 36u2 | 36m2 | 84.00% | |
| 36m2, 36u2 | 36u1 | 91.00% | 87.50% |
| 36m2, 36u2 | 36m1 | 84.00% | |
| | | Overall | 88.00% |

**Table 5-26:** Classification result using all 36 features to discriminate between mixed and upper case word images with the MDA technique.

| Training sets | Test sets | %Correct Linear Classification (MDA) | %Correct Average |
|---|---|---|---|
| 36m2, 36l2 | 36m2 | 77.00% | 77.50% |
| 36m2, 36l2 | 36l2 | 78.00% | |
| 36m2, 36l2 | 36m1 | 77.00% | 75.50% |
| 36m2, 36l2 | 36l1 | 74.00% | |
| | | Overall | 76.50% |

**Table 5-27:** Classification result using all 36 features to discriminate between mixed and lower case word images with the MDA technique.

The overall binary classification using 36 features in MDA technique is 88.25%, 88.00% and 76.50% respectively for classification between upper/lower, upper/mixed and lower/mixed case words. This system can achieve 92.50%, 88.50% and 77.50% correct classification when the test set is the same as training set and 84.00%, 87.50% and 75.50% correct classification when the training set is different to the test set.

Tables 5-28 to 5-30 show the experimental result obtained when using the selected features with the MDA technique detailed in table 3-2 page (3-20). The overall binary classification rate when using selected features in the MDA technique is 80.50%, 85.25% and 75.00% for classification between upper/lower, upper/mixed and lower/mixed case words. This system also achieved 81.50%, 85.50% and 75.50% correct classification when the test set is the same as training set and 79.50%, 85.00% and 74.50% correct classification when the training set is different to the test set.

| Training sets | Test sets | %Correct Linear Classification (MDA) | %Correct Average |
|---|---|---|---|
| 20u2, 2012 | 20u2 | 84.00% | 81.50% |
| 20u2, 2012 | 2012 | 79.00% | |
| 20u2, 2012 | 20u1 | 84.00% | 79.50% |
| 20u2, 2012 | 2011 | 75.00% | |
| | | Overall | 80.50% |

**Table 5-28:** Classification result using the selected 20 features to discriminate between upper and lower case word images with the MDA technique.

| Training sets | Test sets | %Correct Linear Classification (MDA) | %Correct Average |
|---|---|---|---|
| 15m2, 15u2 | 15u2 | 90.00% | 85.50% |
| 15m2, 15u2 | 15m2 | 81.00% | |
| 15m2, 15u2 | 15u1 | 87.00% | 85.00% |
| 15m2, 15u2 | 15m1 | 83.00% | |
| | | Overall | 85.25% |

**Table 5-29:** Classification result using the 15 selected features to discriminate between mixed and upper case word images with the MDA technique.

| Training sets | Test sets | %Correct Linear Classification (MDA) | %Correct Average |
|---|---|---|---|
| 23m2, 2312 | 23m2 | 74.00% | 75.50% |
| 23m2, 2312 | 23612 | 77.00% | |
| 23m2, 2312 | 23m1 | 75.00% | 74.50% |
| 23m2, 2312 | 2311 | 74.00% | |
| | | Overall | 75.00% |

**Table 5-30:** Classification result using the 23 selected features to discriminate between mixed and lower case word images with the MDA technique.

## 5.6 Comparison between the linear and non-linear method for binary case classification

Tables 5-31 and 5-32 summarise the binary classification result using MDA and PNN techniques.

| **Training set is the same as test set** | **Upper/Lower** | | | **Upper/Mixed** | | | **Mixed/Lower** | | |
|---|---|---|---|---|---|---|---|---|---|
| | Dif $\sigma_i$ | Com $\sigma$ | MDA | Dif $\sigma_i$ | Com $\sigma$ | MDA | Dif $\sigma_i$ | Com $\sigma$ | MDA |
| Selected features | 98.00% | 100.00% | **81.50%** | 94.00% | 91.50% | **85.50%** | 96.00% | 96.00% | **75.50** |
| 36 extracted features | 97.50% | 97.00% | **92.50%** | 99.50% | 98.50% | **88.50%** | 97.50% | 97.00% | **77.50%** |

**Table 5-31:** Comparison between the classification results when (i) PNN with using different $\sigma_i$ , (ii) PNN using common $\sigma$ and (iii) MDA techniques, when the training set is the same as test set.

| **Training set is different to the test set** | **Upper/Lower** | | | **Upper/Mixed** | | | **Mixed/Lower** | | |
|---|---|---|---|---|---|---|---|---|---|
| | Dif $\sigma_i$ | Com $\sigma$ | MDA | Dif $\sigma_i$ | Com $\sigma$ | MDA | Dif $\sigma_i$ | Com $\sigma$ | MDA |
| Selected features | 84.50% | 100.00% | **79.50%** | 87.00% | 85.00% | **85.00%** | 77.50% | 77.00% | **74.50%** |
| 36 extracted features | 85.00% | 83.50% | **84.00%** | 88.00% | 84.50% | **87.50%** | 81.00% | 73.00% | **75.50%** |

**Table 5-32:** Comparison between the classification results when (i) PNN using different $\sigma_i$ , (ii) PNN using common $\sigma$ and (iii) MDA techniques, when the training set is different to the test set.

### 5.6.1 36 Extracted features

The experimental results given in table 5-31 and 5-32 (page 5-33) show that the classification rate using the PNN system achieved an improvement of 1.00%, 0.50% and 5.50% with different $\sigma_i$ and a decrease of 0.50%, 3.00% and 2.50% with common $\sigma$ when compared to the MDA technique for classification between lower/upper, upper/mixed and lower/mixed case words where the test set is different to the training set.

The experimental results given in table 5-31 also show that the classification rate using the PNN system achieved an improvement of 5.00%, 11.00% and 20.00% with different $\sigma_i$ and an improvement of 4.50%, 10.00% and 9.50% with common $\sigma$ compared to the MDA technique for classification between lower/ upper, upper/mixed and lower/mixed case words where the test set is the same as the training set.

The experimental results given shown in tables 5-31 and 5-32 show that an increase of 1.50% and 3.50% and 8.00% is achieved when using different $\sigma_i$ in comparison to using the common $\sigma$ for classification between lower/upper, mixed/upper and mixed/lower respectively when the training set is different with the test set. An improvement of 0.50%, 1.00% and 0.50% is achieved when using different $\sigma_i$ in comparison to using the common $\sigma$ when the test set is the same as training set. Overall then, these experiments show that using different $\sigma_i$ with 36 extracted common features helps the classifier to discriminate between lower/upper, upper/mixed and lower/mixed case words compared to using a common $\sigma$ with the 36 features.

### 5.6.2 Selected features

The experimental results given in table 5-32 show that a classification using the PNN system achieved an improvement of 5.00%, 2.00% and 3.00% with different $\sigma_i$ and an improvement of 20.50%, 0% and 2.50% with common $\sigma$ compared to the MDA technique for classification between lower/upper, upper/mixed and lower/mixed case words when the test set is different to the training set.

The experimental result given in table 5-31 show that classification using PNN achieved an improvement of 16.50%, 8.50% and 21.50% using different $\sigma_i$ and an improvement of 18.50%, 6.50% and 20.5% using common $\sigma$ compared to the MDA technique for classification between lower/upper, upper/mixed and lower/mixed case words when the training set is the same as the test set.

Overall then, these experiments show that when using the selected features the best classification result for upper/lower case word images is by using common $\sigma$. However using different $\sigma_i$ values does help the classifier to better discriminate between upper/mixed and lower/mixed.

## 5.7 Conclusion

Two methods for the case classification of the word images are described (MDA and PNN) and a comparison between these two methods is presented. The experimental results using MDA and PNN techniques with different $\sigma_i$ and common $\sigma$ show that the PNN technique using different $\sigma_i$ values gives the best classification result and that the PNN technique with common $\sigma$ gives nearly the same classification result as MDA technique.

Selected features for classification of lower/upper case words using PNN with common $\sigma$ gave the best classification result but all 36 features are needed for classification of lower/mixed and upper/mixed case words to give a better result. More discussion about zoning information is given in Appendix B.

# 6. PREDICTION OF LEGIBILITY BASED ON EXISTING RECOGNISER

## 6.1 Introduction

Many methods have been developed for handwriting recognition and, in general, they all attempt to deal with poorly written handwriting [HAMANAKA00]. Indeed, Coates, Baird and Fateman [COATES01] have shown experimentally that there are a variety of images, which though legible to human readers are illegible to several of the best present day optical character recognition systems. In this work it has therefore been hypothesised that one way of helping cursive script recognition would be to detect writing style prior to the recognition stage in order to choose the best recogniser for the given writing style. In this work the concept of style classification is introduced and the various aspects of its definition in quantitative terms are discussed. To provide a starting point, style has been defined in terms of recogniser specific legibility. In this way the best recogniser could be selected for a given style of writing using a prediction of legibility based on a given recogniser's previous performance. This research therefore focuses on the problem of classifying word images as legible, illegible or middling prior to the recognition stage. An independent handwriting style classifier has been designed that, in principal, can be used to select the best recognizer for a given style of writing. For this purpose a definition of recogniser specific handwriting legibility has been defined and a method has been implemented that can predict this legibility [EBADIAN01].

In chapter 5 a MDA and a PNN based on the Bayes strategy technique were proposed for case classification. In this chapter both methods are applied to the task of classifying words into legible, illegible or middling prior to the recognition stage. A comparison between the two classification techniques can thus be given.

## 6.2 Definition of Legibility

Up until now handwriting legibility has been defined purely in human terms. However, since the ability of a machine-based recogniser differs significantly from that of a human being [COATES01], any definition of legibility should be based on the recognition system. Of course, similar to that of a human being, the definition of legibility is a debatable issue. However at the time of writing no reference to a machine based definition of legibility has been found in the literature, which is probably not surprising considering the novelty of this concept.

Our definition of handwritten legibility has therefore been based on our existing recogniser's (HVBC) performance [SHERKAT99]. HVBC is a holistic word level recogniser that uses three features namely, Holes, Vertical bars and Cups. However, this definition of legibility can be extended to any available recogniser. Figure 6-1 shows that almost all correct words are located in a top 10 position. Thus legible words are defined as those that are likely to be placed in the top 10 of the correct word list with a score of 75% or greater. Illegible words are defined as those that would produce a list containing the correct word, any where in the word list with a score of less than 45%. Middle words (those between legible and illegible) are then defined as those that would produce a list containing the correct word with a score of 45% to 75%.

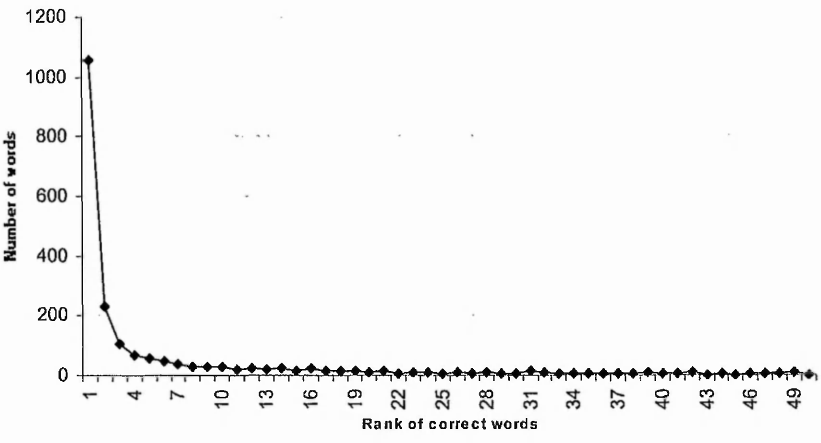



**Figure 6-1:** All correct words regardless of rank.

These thresholds have been selected and merely provide a starting point. They can be changed depending on the application in which they are to be used [MADHVANATH01]. The following experiments show the results of binary style classification followed by triple style classification. The style classification technique was applied on our existing data set, which consists of scanned images obtained from eighteen writers each containing 150 words at $200 \times 100$ dpi resolution. Initially the system is trained on the LEGTRn (legible training words), ILLEGTRn (illegible training words) and MiddleTRn (middle training words) sets containing all 2456 words in the training set. The classification system was then tested with (1) the same data set: LEGTRn, ILLEGTRn and MiddleTRn and (2) a different data set, LEGTEn (legible test words), ILLEGTEn (illegible test words) and MiddleTEn (middle test words). This latter set containing 518 words, Note that *n* in the name of the data sets (LEGTRn, ILLEGTRn, MiddleTRn, LEGTEn, ILLEGTEn and MiddleTEn) shows the number of features and TR and TE indicate the training and test sets respectively. In this chapter the x-axis, y-axis and z-axis lines in figures 6-2 to 6-7 (pages 6-5, 6-6, 6-8 and 6-9), figures 6-10 (page 6-13) and 6-11 (page 6-14) indicate the number of segmented sigma's range and the estimated error in each region respectively (see chapter 4). Sigma's range and error function are shown in the tables under each figure.

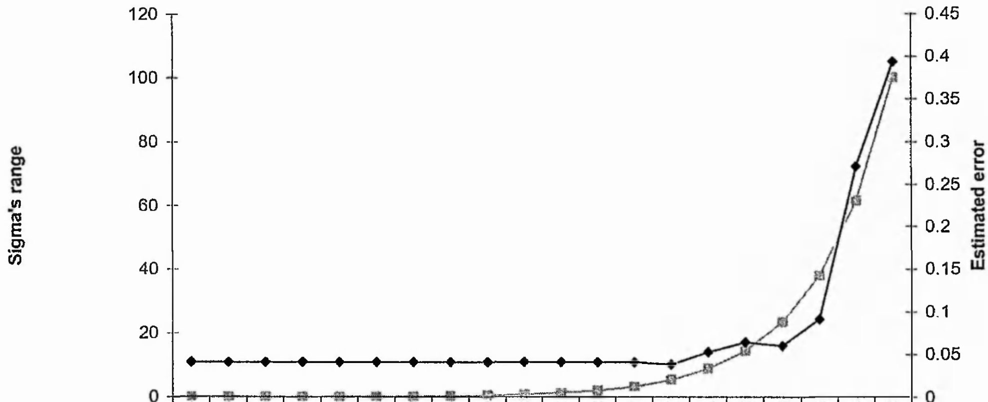
## 6.3 PNN Style Classifier Using a Common $\sigma$

### 6.3.1 Binary Classification

Tables 6-1 to 6-6 (pages 6-6, 6-7, 6-9 and 6-10) show the two class (binary) classification results obtained when using non-linear classification (PNN) techniques based on the selected values of common $\sigma$ . The first column in these tables shows the samples that were used as the training data set whilst the second column shows the samples that were used as a test set. The third column shows the correct classification results obtained when using a non-linear classification (PNN) technique with common $\sigma$ using all of the 36 features (table 6-1 to 6-3) or selected features (tables 6-4 to 6-6). The fourth column shows the average of correct classification results when the system was tested with seen or unseen data. The last row shows the average classification result for all with common $\sigma$ .

#### 6.3.1.1 Experimental Results and Analysis Using 36 Extracted Features

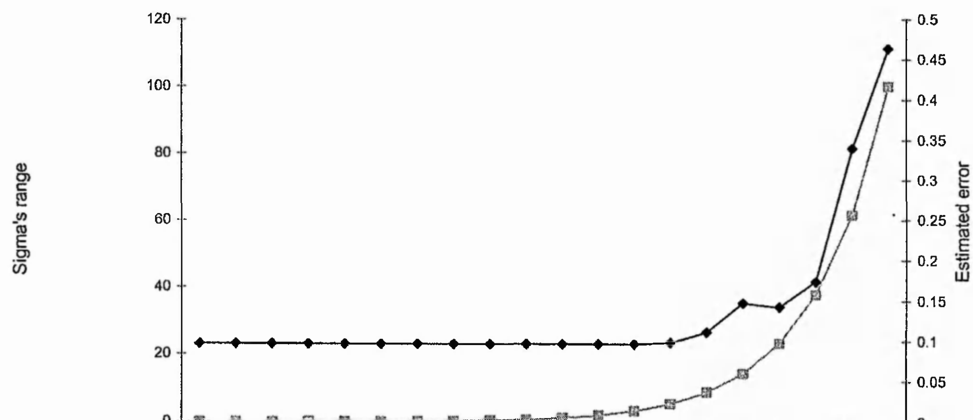
The results shown in figures 6-2 to 6-4 indicate that the best value of $\sigma$ lies within the 3.3598 to 8.8587 interval. It is calculated as 5.47436 for the case of legible and illegible words with an error rate of 0.03836. The $\sigma$ value lies within the 0.01 to 5.4556 interval and is calculated as 0.01385 for middle and illegible words with an error rate of 0.09580. For classification between legible and middle words the $\sigma$ value lies within the 2.6367 to 8.85870 interval and is calculated as 7.11064 with an error rate of 0.42720.




| | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Sigma's range | 0.01 | 0.02 | 0.03 | 0.04 | 0.07 | 0.11 | 0.18 | 0.3 | 0.48 | 0.78 | 1.27 | 2.07 | 3.36 | 5.46 | 8.86 | 14.4 | 23.4 | 37.9 | 61.6 | 100 |
| Error function | 0.04 | 0.04 | 0.04 | 0.04 | 0.04 | 0.04 | 0.04 | 0.04 | 0.04 | 0.04 | 0.04 | 0.04 | 0.04 | 0.04 | 0.05 | 0.06 | 0.06 | 0.09 | 0.27 | 0.39 |

Number of segmented sigma's range

**Figure 6-2:** Error estimation of common $\sigma$ for a classification between legible and illegible handwriting using 36 extracted features ($\sigma$ =5.47436).




| | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Sigma's range | 0.01 | 0.02 | 0.03 | 0.04 | 0.07 | 0.11 | 0.18 | 0.3 | 0.48 | 0.78 | 1.27 | 2.07 | 3.36 | 5.46 | 8.86 | 14.4 | 23.4 | 37.9 | 61.6 | 100 |
| Error function | 0.1 | 0.1 | 0.1 | 0.1 | 0.1 | 0.1 | 0.1 | 0.1 | 0.1 | 0.1 | 0.1 | 0.1 | 0.1 | 0.1 | 0.11 | 0.15 | 0.14 | 0.17 | 0.34 | 0.46 |

Number of segmented sigma's region

**Figure 6-3:** Error estimation of common $\sigma$ for a classification between middle and illegible handwriting using 36 extracted features ($\sigma$ =0.01386).

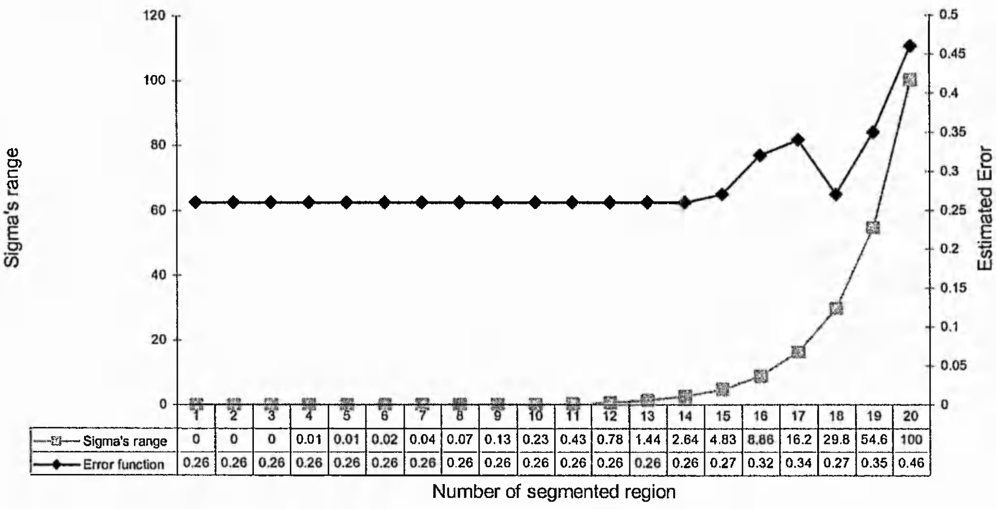



**Figure 6-4:** Error estimation of common $\sigma$ for a classification between middle and legible handwriting using 36 extracted features ($\sigma$ =7.11064).

| Training set | Test set | % Correct Classification result (common $\sigma$) | %Correct Average |
|---|---|---|---|
| LEGTR36, ILLEGTR36 | LEGTR36 | 99.00% | 99.50% |
| LEGTR36, ILLEGTR36 | ILLEGTR36 | 100.00% | |
| LEGTR36, ILLEGTR36 | LEGTE36 | 69.00% | 79.50% |
| LEGTR36, ILLEGTR36 | ILLEGTE36 | 90.00% | |
| | | Overall | 89.50% |

**Table 6-1:** Classification results using 36 extracted features to discriminate between legible and illegible handwriting using common $\sigma$ ($\sigma$ =5.47436).

| Training set | Test set | % Correct Classification result (common $\sigma$) | %Correct Average |
|---|---|---|---|
| LEGTR36, MiddleTR36 | LEGTR36 | 100.00% | 99.50% |
| LEGTR36, MiddleTR36 | MiddleTR36 | 99.00% | |
| LEGTR36, MiddleTR36 | LEGTE36 | 81.00% | 65.50% |
| LEGTR36, MiddleTR36 | MiddleTE36 | 50.00% | |
| | | Overall | 82.50% |

**Table 6-2:** Classification results using 36 extracted features to discriminate between legible and middle handwriting using common $\sigma$ ($\sigma$ =7.11064).

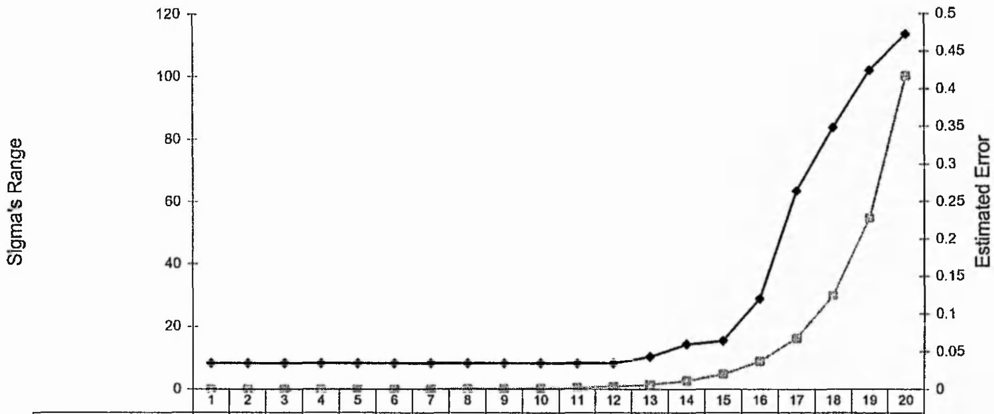
| Training set | Test set | % Correct Classification result (common $\sigma$) | %Correct Average |
|---|---|---|---|
| MiddleTR36, ILLEGTR36 | MiddleTR36 | 99.00% | 99.50% |
| MiddleTR36, ILLEGTR36 | ILLEGTR36 | 100.00% | |
| MiddleTR36, ILLEGTR36 | MiddleTE36 | 52.00% | 76.00% |
| MiddleTR36, ILLEGTR36 | ILLEGTE36 | 100.00% | |
| | | Overall | 87.75% |

**Table 6-3:** Classification result using 36 extracted features to discriminate between middle and illegible handwriting using common $\sigma$ ($\sigma$ =0.01386).

Tables 6-1, 6-2 and 6-3 show that the average classification result is 89.50% , 82.50% and 87.75% when classifying between legible/illegible, legible/middle and illegible/middle word images respectively using 36 extracted features. The system can also achiev 99.50%, 99.50% and 99.50% correct classification when the test set is the same as the training set and 79.50%, 65.50% and 76.00% correct classification when the test set is different to the training set.

### 6.3.1.2 Experimental Results and Analysis Using The Selected Features

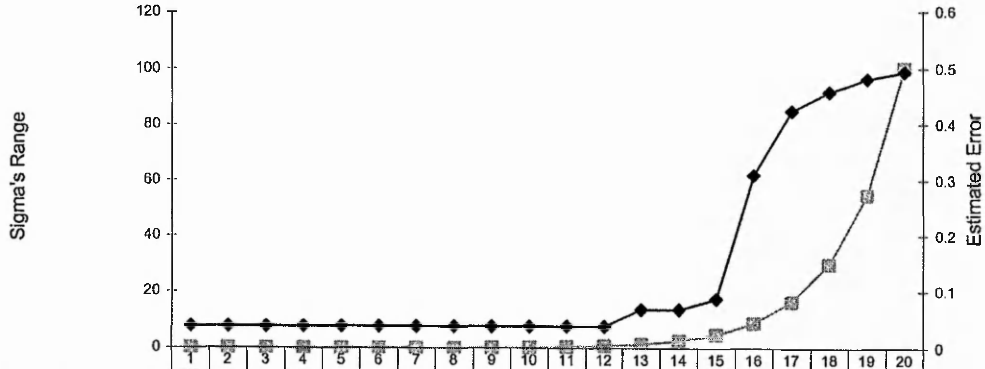
The results shown in figures 6-5 to 6-7 indicate that the best value of $\sigma$ lies within the 0.0001 to 0.001 intervals for legible/illegible word style classification. It is calculated as 0.00066 with an error rate of 0.03445. The common $\sigma$ value lies within the 0.0001 to 0.001 interval for middle/illegible word classification and is calculated as 0.0001 with an error rate of 0.0956. For legible and middle words the common $\sigma$ value lies within the 0.0001 to 0.01 interval and is calculated as 0.0015 with an error rate of 0.3785.




| | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Sigma's Range | 0 | 0 | 0 | 0.01 | 0.01 | 0.02 | 0.04 | 0.07 | 0.13 | 0.23 | 0.43 | 0.78 | 1.44 | 2.64 | 4.83 | 8.86 | 16.2 | 29.8 | 54.6 | 100 |
| Error Estimation | 0.03 | 0.03 | 0.03 | 0.03 | 0.03 | 0.03 | 0.03 | 0.03 | 0.03 | 0.03 | 0.03 | 0.03 | 0.04 | 0.06 | 0.06 | 0.12 | 0.26 | 0.35 | 0.42 | 0.47 |

Number of Segmented Regions

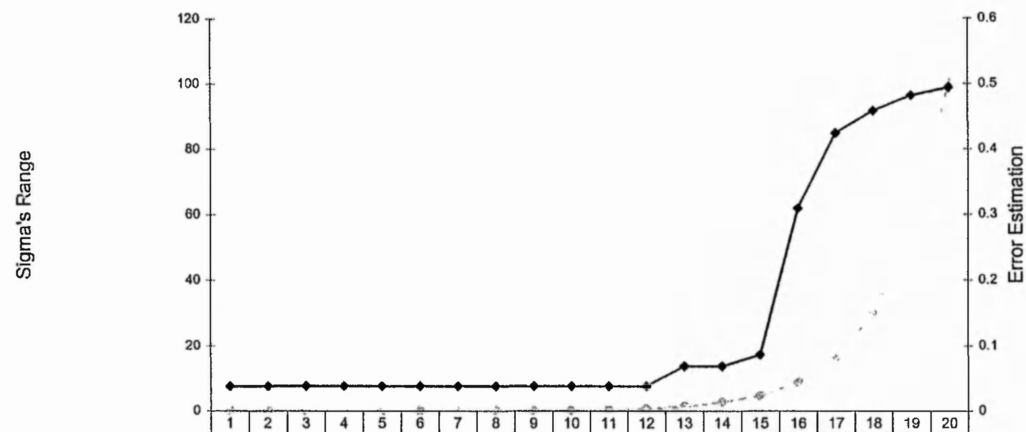
**Figure 6-5:** Error estimation of common $\sigma$ for a classification between illegible and legible handwriting using 16 selected features ($\sigma$ =0.00066).




| | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Sigma's Range | 0 | 0 | 0 | 0.01 | 0.01 | 0.02 | 0.04 | 0.07 | 0.13 | 0.23 | 0.43 | 0.78 | 1.44 | 2.64 | 4.83 | 8.86 | 16.2 | 29.8 | 54.6 | 100 |
| Error Rate | 0.04 | 0.04 | 0.04 | 0.04 | 0.04 | 0.04 | 0.04 | 0.04 | 0.04 | 0.04 | 0.04 | 0.04 | 0.07 | 0.07 | 0.09 | 0.31 | 0.42 | 0.46 | 0.48 | 0.49 |

Number of Segmented Regions

**Figure 6-6:** Error estimation of common $\sigma$ for a classification between middle and legible handwriting using 15 selected features ($\sigma$ =0.00150).




| | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Sigma's Range | 0 | 0 | 0 | 0.01 | 0.01 | 0.02 | 0.04 | 0.07 | 0.13 | 0.23 | 0.43 | 0.78 | 1.44 | 2.64 | 4.83 | 8.86 | 16.2 | 29.8 | 54.6 | 100 |
| Error Estimation | 0.04 | 0.04 | 0.04 | 0.04 | 0.04 | 0.04 | 0.04 | 0.04 | 0.04 | 0.04 | 0.04 | 0.04 | 0.07 | 0.07 | 0.09 | 0.31 | 0.42 | 0.46 | 0.48 | 0.49 |

Number of segmented regions

**Figure 6-7:** Error estimation of common $\sigma$ for a classification between middle and illegible handwriting using 13 selected features ($\sigma$ =0.0001).

| Training set | Test set | % Correct Classification result (common $\sigma$) | %Correct Average |
|---|---|---|---|
| LEGTR16, ILLEGTR16 | LEGTR16 | 99.00% | 99.50% |
| LEGTR16, ILLEGTR16 | ILLEGTR16 | 100.00% | |
| LEGTR16, ILLEGTR16 | LEGTE16 | 93.00% | 78.00% |
| LEGTR16, ILLEGTR16 | ILLEGTE16 | 63.00% | |
| | | Overall | 87.75% |

**Table 6-4:** Classification result using 16 selected features to discriminate between legible and illegible handwriting using common $\sigma$ ($\sigma$ =0.00066).

| Training set | Test set | % Correct Classification result (common $\sigma$) | %Correct Average |
|---|---|---|---|
| LEGTR15, MiddleTR15 | LEGTR15 | 100.00% | 100.00% |
| LEGTR15, MiddleTR15 | MiddleTR15 | 100.00% | |
| LEGTR15, MiddleTR15 | LEGTE15 | 88.00% | 70.50% |
| LEGTR15, MiddleTR15 | MiddleTE15 | 63.00% | |
| | | Overall | 85.25% |

**Table 6-5:** Classification result using 15 selected features to discriminate between legible and middle handwriting using common $\sigma$ ($\sigma$ =0.00150).

| Training set | Test set | % Correct Classification result (common $\sigma$) | %Correct Average |
|---|---|---|---|
| MiddleTR13, ILLEGTR13 | MiddleTR13 | 99% | 99.5% |
| MiddleTR13, ILLEGTR13 | ILLEGTR13 | 100% | |
| MiddleTR13, ILLEGTR13 | MiddleTE13 | 60% | 80% |
| MiddleTR13, ILLEGTR13 | ILLEGTE13 | 100% | |
| | | Overall | 89.75% |

**Table 6-6:** Classification result using 13 selected features to discriminate between illegible and middle handwriting using common $\sigma$ ($\sigma$ =0.0001).

Tables 6-4, 6-5 and 6-6 show that the average classification result is 88.75%, 85.25% and 89.75% when classifying between legible/illegible, legible/middle and illegible/middle word images respectively using selected features. This can be broken down into 99.50%, 100.00% and 99.50% correct classification when the test set is the same as the training set and 78.00%, 70.50% and 80% correct classification when the test set is different to the training set.

#### 6.3.1.3 Comparison Between Using The Selected and 36 Extracted Features

Figures 6-8 and 6-9 summarise the results from tables 6-1 to 6-6 (pages 6-6, 6-7, 6-9 and 6-10). Figure 6-8 shows the classification result when the training set is the same as the test set and figure 6-9 shows the classification result when the training set is different to the test set.

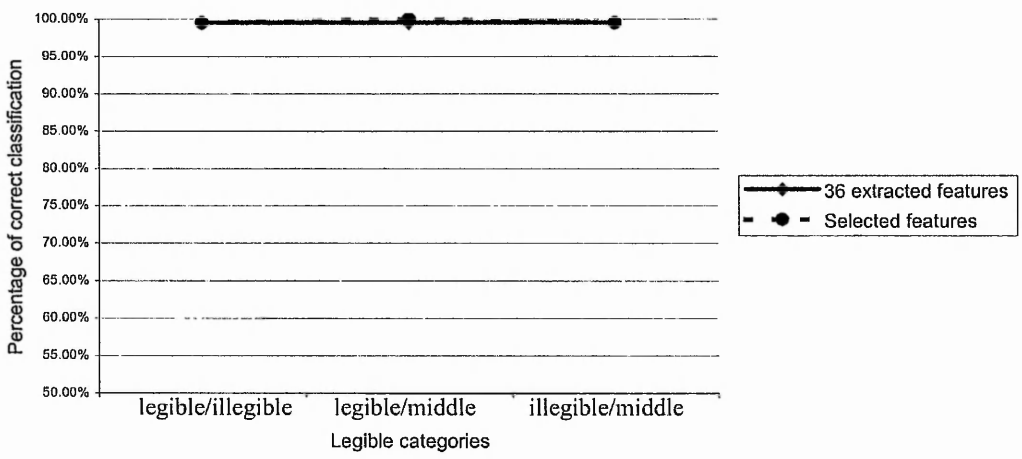



**Figure 6-8:** Comparison between the selected and 36 extracted features using common $\sigma$ for seen data.

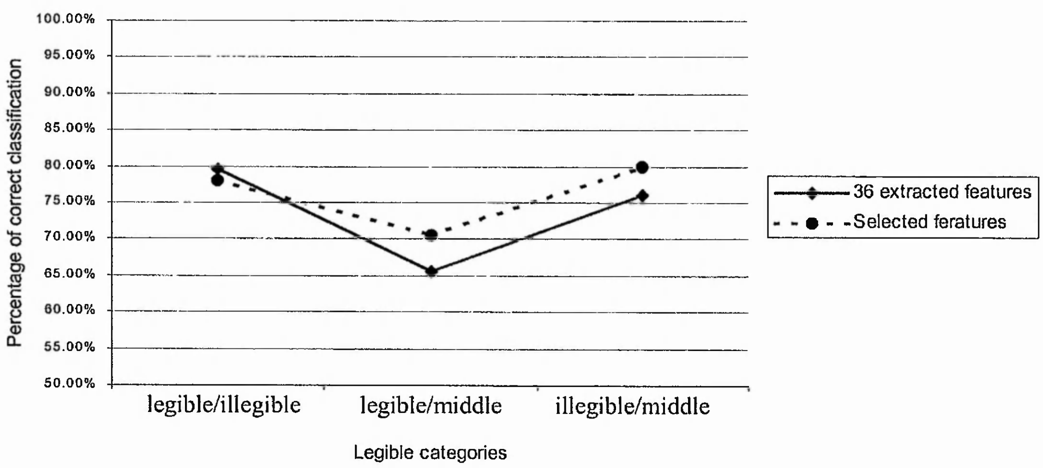



**Figure 6-9:** Comparison between the selected and 36 extracted features using common $\sigma$ for unseen data.

Figure 6-8 shows that when classifying between legible/illegible, legible/middle and illegible/middle there is virtually no difference using 36 features and the selected features when the test set is the same as the training set.

Figure 6-9 shows that when classifying between legible/illegible, legible/middle and illegible/middle an improvement of 1.50%, 5.00% and 4.00% can be achieved by using the selected features rather than all 36 features when the test set is different with the training set.

### 6.3.2 Triple Classification

Tables 6-7 and 6-8 (page 6-13 and 6-15) gives the results for the 3 class data sets. The first column shows the samples that were used as the training data set whist the second column shows the samples that were used as the test set. The third column shows the correct classification results obtained when using the non-linear classification technique with common $\sigma$ using (i) all 36 features (table 6-13) and (ii) selected features (table 6-14). The fourth, fifth, sixth and seventh columns show the misclassification results in each category and the average classification result for seen and unseen data. The last row shows the overall classification result for all with common $\sigma$ .

#### 6.3.2.1 Experimental Results and Analysis Using 36 Extracted Features

For three class style classification the best common $\sigma$ value is 0.001, which lies within the 0.001 and 0.0018 interval, with an error rate of 0.33379. The details are shown in figure 6-10.

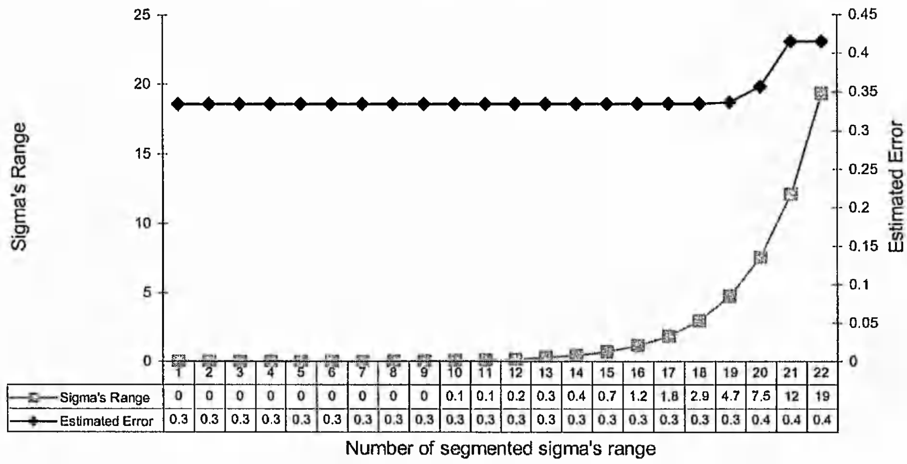



**Figure 6-10:** Error estimation of common $\sigma$ for a classification between legible, illegible and middle handwriting using 36 extracted features ($\sigma$ =0.001).

## %Misclassification words

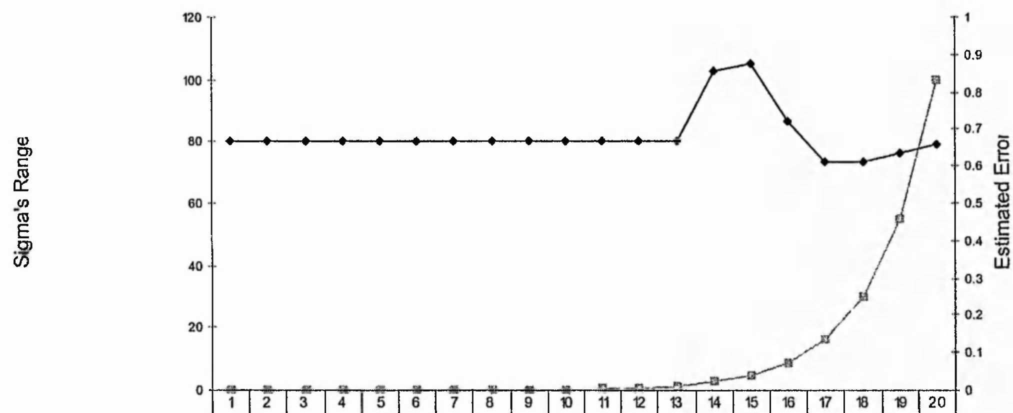
| Training files | Test files | %Correct non-linear (PNN) | As Legible | As Illegible | As Middle | %Correct Average |
|---|---|---|---|---|---|---|
| LEGTR36, ILLEGTR36,MiddleTR36 | LEGTR36 | 100.00% | - | 0 | 0 | 99.67% |
| LEGTR36, ILLEGTR36,MiddleTR36 | ILLEGTR36 | 100.00% | 0 | - | - | |
| LEGTR36, ILLEGTR20,MiddleTR36 | MiddleTR36 | 99.00% | 1.00% | 0 | - | |
| LEGTR36, ILLEGTR36,MiddleTR36 | LEGTE36 | 72.00% | - | 10.00% | 18.00% | 67.33% |
| LEGTR36, ILLEGTR36,MiddleTR36 | ILLEGTE36 | 83.00% | 17.00% | - | 0 | |
| LEGTR36, ILLEGTR36,MiddleTR36 | MiddleTE36 | 47.00% | 51.00% | 2.00% | - | |
| | | | | | Overall | 83.50% |

**Table 6-7:** Classification results using 36 features to discriminate between legible, illegible and middle handwriting word images using common $\sigma$ ($\sigma$ =0.001).

The experimental results given in table 6-7 show that a classifier based on the probabilistic neural network (PNN) using a common $\sigma$ value of 0.001 can achieve an overall correct style classification of 67.30% when the test set is different to the training set. The system can also be seen to achieve a 99.70% correct classification when the test set is the same as the training set. This gives an overall correct classification of 83.50% for the three classes.

### 6.3.2.2 Experimental Results and Analysis Using The Selected Features

For three class style classification using 20 features the best common $\sigma$ value is 22.22464, which lies within the 16.23780 and 29.76350 interval, with an error rate of 0.60362. The details are shown in figure 6-11.




| | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Sigma's Range | 0 | 0 | 0 | 0.01 | 0.01 | 0.02 | 0.04 | 0.07 | 0.13 | 0.23 | 0.43 | 0.78 | 1.44 | 2.64 | 4.83 | 8.86 | 16.2 | 29.8 | 54.6 | 100 |
| Estimated Error | 0.67 | 0.67 | 0.67 | 0.67 | 0.67 | 0.67 | 0.67 | 0.67 | 0.67 | 0.67 | 0.67 | 0.67 | 0.67 | 0.85 | 0.88 | 0.72 | 0.61 | 0.61 | 0.64 | 0.66 |

Number of segmented sigma's region

**Figure 6-11:** Error estimation of common $\sigma$ for a classification between legible, illegible and middle words using 20 extracted features ($\sigma$ = 22.22464).

| Training files | Test files | %Correct non-linear (PNN) | %Misclassification words: AS Legible | %Misclassification words: As Illegible | %Misclassification words: As Middle | %Correct Average |
|---|---|---|---|---|---|---|
| LEGTR20, ILLEGTR20,MiddleTR20 | LEGTR20 | 88.00% | - | 7.00% | 5.00% | |
| LEGTR20, ILLEGTR20,MiddleTR20 | ILLEGTR20 | 71.00% | 26.00% | - | 3.00% | 66.00% |
| LEGTR20, ILLEGTR20,MiddleTR20 | Middle TR20 | 39.00% | 46.00% | 15.00% | - | |
| LEGTR20, ILLEGTR20,MiddleTR20 | LEGTE20 | 76.00% | - | 5.00% | 19.00% | |
| LEGTR20, ILLEGTR20,MiddleTR20 | ILLEGTE20 | 42.00% | 45.00% | - | 13.00% | 46.00% |
| LEGTR20, ILLEGTR20,MiddleTR20 | Middle TE20 | 20.00% | 44.00% | 36.00% | - | |
| | | | | | Overall | 56.00% |

**Table 6-8:** Classification result using 20 selected features to discriminate between legible, illegible and middle handwriting word images using common $\sigma$ ($\sigma$ =22.22464).

The experimental results given in table 6-8 show that a classifier based on probabilistic neural network (PNN) using a common $\sigma$ value of 22.22464 can achieve an overall correct style classification of 46.00% when the test set is different to the training set. This result is poor and the system achieves 76.00% correct classification for legible words but it is very poor for classifying illegible and middle. It is suggested to introduce some new features to improve this classification. The system can also be seen to achieve a 66.00% correct classification when the test set is the same as the training set using 20 selected features. This gives an overall 56.00% correct classification for the three classes.

#### 6.3.2.3 Comparison Between Using The Selected and 36 Extracted Features

Figures 6-12 and 6-13 summarise the results of tables 6-7 and 6-8 (page 6-13 and 6-15). Figure 6-12 shows classification result when the training set is the same as the test set and figure 6-13 shows the classification results when the training set is different to the test set. In these figures 1,2 and 3 on the axis line indicates the legible, illegible and middle classifications respectively.

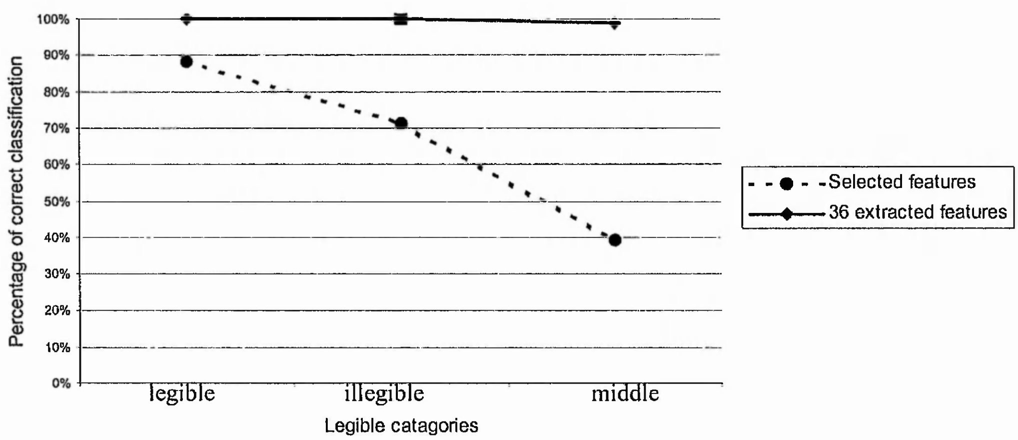



**Figure 6-12:** Comparison between using the selected and 36 extracted features using with common $\sigma$ for seen data.

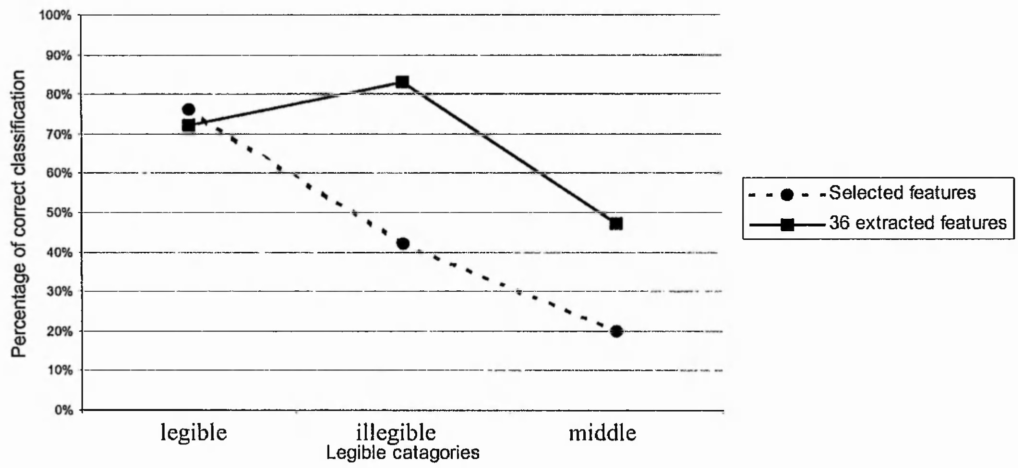



**Figure 6-13:** Comparison between using the selected and 36 extracted features with common $\sigma$ for unseen data.

Figure 6-12 shows that in triple classification an improvement of 12.00%, 29.00% and 60.00% can be achieved for legible, illegible and middle by using all 36 features rather than the of 20 selected features when the test set is the same as training set.

Figure 6-13 also shows that in triple classification an decrease of 4% and an improvement 41.00% and 27.00% can be achieved for legible, illegible and middle classification by using all 36 features rather than the 20 selected features when the test set is different with the training set.

## 6.4 PNN Style Classifier Using Different $\sigma_i$

### 6.4.1 Binary Classification

Tables 6-10 to 6-12 (page 6-19 and 6-20) and 6-14 to 6-16 (page 6-22) show the classification results obtained when using a non-linear classification (PNN) technique with the different values of $\sigma_i$ $(i = 1,2,\cdots,36)$. The first column in these tables show the samples that were used as the training data set whilst the second column show the correct classification result obtained when using a non-linear classification (PNN) technique with different $\sigma_i$ using (i) all 36 features (tables 6-10 to 6-12) or (ii) selected features (tables 6-14 to 6-16).

### 6.4.1.1 Experimental Results and Analysis Using 36 Extracted Features

Table 6-9 shows different value of $\sigma_i$ in binary classification using 36 features.

| Legible/ Illegible $\sigma_i$ in LEG class | Legible/ Illegible $\sigma_i$ in Illegible class | Middle/ Illegible $\sigma_i$ in Middle class | Middle/ Illegible $\sigma_i$ in Illegible class | Legible/ Middle $\sigma_i$ in legible Class | Legible/ Middle $\sigma_i$ in Middle Class |
|---|---|---|---|---|---|
| 62.332247 | 0 | 1.426 | 1.45026 | 0.031938 | 0.033382 |
| 61.767636 | 0 | 1.426 | 1.45026 | 0.031938 | 0.033382 |
| 56.440787 | 3.775868 | 1.426 | 1.45026 | 0.031938 | 0.033382 |
| 57.151096 | 3.304562 | 1.426 | 1.45026 | 0.031938 | 0.033382 |
| 57.389402 | 1.357227 | 1.426 | 1.45026 | 0.031938 | 0.033382 |
| 62.366072 | 0 | 1.426 | 1.45026 | 0.031938 | 0.033382 |
| 59.681608 | 1.114038 | 1.426 | 1.45026 | 0.031938 | 0.033382 |
| 60.975208 | 0 | 1.426 | 1.45026 | 0.031938 | 0.033382 |
| 59.748686 | 2.086025 | 1.426 | 1.45026 | 0.031938 | 0.033382 |
| 62.580416 | 0 | 1.426 | 1.45026 | 0.031938 | 0.033382 |
| 62.638468 | 0 | 1.426 | 1.45026 | 0.031938 | 0.033382 |
| 60.474392 | 0 | 1.426 | 1.45026 | 0.031938 | 0.033382 |
| 61.386263 | 0 | 1.426 | 1.45026 | 0.031938 | 0.033382 |
| 61.388439 | 0 | 1.426 | 1.45026 | 0.031938 | 0.033382 |
| 38.042082 | 22.221027 | 1.426 | 1.45026 | 0.031938 | 0.033382 |
| 46.292339 | 16.076158 | 1.426 | 1.45026 | 0.031938 | 0.033382 |
| 29.98878 | 14.277587 | 1.426 | 1.45026 | 0.031938 | 0.033382 |
| 28.685453 | 29.923836 | 1.426 | 1.45026 | 0.031938 | 0.033382 |
| 46.425980 | 13.598515 | 1.426 | 1.45026 | 0.031938 | 0.033382 |
| 14.344564 | 46.641445 | 1.426 | 1.45026 | 0.031938 | 0.033382 |
| 35.742673 | 23.736243 | 1.426 | 1.45026 | 0.031938 | 0.033382 |
| 41.023238 | 17.073676 | 1.426 | 1.45026 | 0.031938 | 0.033382 |
| 23.631395 | 36.951962 | 1.426 | 1.45026 | 0.031938 | 0.033382 |
| 52.113523 | 6.516233 | 1.426 | 1.45026 | 0.031938 | 0.033382 |
| 62.667461 | 0 | 1.426 | 1.45026 | 0.031938 | 0.033382 |
| 52.285391 | 5.920513 | 1.426 | 1.45026 | 0.031938 | 0.033382 |
| 62.022171 | 0 | 1.426 | 1.45026 | 0.031938 | 0.033382 |
| 51.143473 | 13.117231 | 1.426 | 1.45026 | 0.031938 | 0.033382 |
| 62.613546 | 0 | 1.426 | 1.45026 | 0.031938 | 0.033382 |
| 36.051134 | 24.204868 | 1.426 | 1.45026 | 0.031938 | 0.033382 |
| 46.400312 | 15.415807 | 1.426 | 1.45026 | 0.031938 | 0.033382 |
| 35.700545 | 21.019975 | 1.426 | 1.45026 | 0.031938 | 0.033382 |
| 49.205683 | 14.911681 | 1.426 | 1.45026 | 0.031938 | 0.033382 |
| 44.872470 | 18.381896 | 1.426 | 1.45026 | 0.031938 | 0.033382 |
| 46.035568 | 8.510058 | 1.426 | 1.45026 | 0.031938 | 0.033382 |
| 45.938374 | 11.682932 | 1.426 | 1.45026 | 0.031938 | 0.033382 |

**Table 6-9:** Thirty-six different $\sigma_i$ for each legible, illegible and middle class using 36 extracted features.

Table 6-9 shows the best values of different $\sigma_i$ obtained for each legible, illegible and middle style classification with an error rate of 0.16901, 0.08743 and 0.15428 for legible/illegible, middle/illegible and legible/middle respectively. The first and second columns show the different $\sigma_i$ values for legible/illegible classification, the third and fourth columns show the different $\sigma_i$ values for middle/illegible classification and the fifth and sixth columns show the different $\sigma_i$ values for legible/middle classification.

| Training set | Test set | % Correct Classification (different $\sigma_i$) | % Correct Average |
|---|---|---|---|
| LEGTR36, ILLEGTR36 | LEGTR36 | 99.00% | 99.50% |
| LEGTR36, ILLEGTR36 | ILLEGTR36 | 100.00% | |
| LEGTR36, ILLEGTR36 | LEGTE36 | 90.00% | 86.50% |
| LEGTR36, ILLEGTR36 | ILLEGTE36 | 83% | |
| | | Overall | 93.00% |

**Table 6-10:** Classification result using 36 extracted features to discriminate between illegible and legible handwriting using different $\sigma_i$.

| Training set | Test set | % Correct Classification (different $\sigma_i$) | % Correct Average |
|---|---|---|---|
| LEGTR36, MiddleTR36 | LEGTR36 | 100.00% | 99.50% |
| LEGTR36, MiddleTR36 | MiddleTR36 | 99.00% | |
| LEGTR36, MiddleTR36 | LEGTE36 | 81.00% | 65.50% |
| LEGTR36, MiddleTR36 | MiddleTE36 | 50.00% | |
| | | Overall | 82.50% |

**Table 6-11:** Classification result using 36 extracted features to discriminate between middle and legible handwriting using different $\sigma_i$.

| Training set | Test set | % Correct Classification (different $\sigma_i$) | % Correct Average |
|---|---|---|---|
| MiddleTR36, ILLEGTR36 | MiddleTR36 | 99.00% | 99.50% |
| MiddleTR36, ILLEGTR36 | ILLEGTR36 | 100.00% | |
| MiddleTR36, ILLEGTR36 | MiddleTE36 | 98.00% | 90.50% |
| MiddleTR36, ILLEGTR36 | ILLEGTE36 | 83.00% | |
| | | Overall | 95.00% |

**Table 6-12:** Classification result using 36 extracted features to discriminate between middle and illegible handwriting using different $\sigma_i$.

Tables 6-10, 6-11 and 6-12 show that the overall classification results are 93.00%, 82.50% and 95.00% correct classification when classifying legible/middle, illegible/middle and legible/illegible, and handwriting word images respectively. These can be broken down into 99.50%, 99.50% and 99.50% correct classification when the test set is the same as the training set and 86.00%, 65.50% and 90.50% correct classification when the test set is different to the training set.

### 6.4.1.2 Experimental Results and Analysis Using The Selected Features

Table 6-13 shows different value of $\sigma_i$ in binary classification using 36 features.

| Legible/ Illegible dif $\sigma_i$ in LEG class | Legible/Illegible dif $\sigma_i$ in illegible class | Middle/ Illegible dif $\sigma_i$ in Middle class | Middle/Illegible dif $\sigma_i$ in Illegible class | Legible/ Middle dif $\sigma_i$ in Legible class | Legible/ Middle dif $\sigma_i$ in Middle class |
|---|---|---|---|---|---|
| 0.001778 | 0.002 | 0.00097 | 0.000103 | 0.555113 | 0.00000001 |
| 0.001778 | 0.002 | 0.00097 | 0.000103 | 0.555113 | 0.00000001 |
| 0.001778 | 0.002 | 0.00097 | 0.000103 | 0.555113 | 0.00000001 |
| 0.001778 | 0.002 | 0.00097 | 0.000103 | 0.555113 | 0.00000001 |
| 0.001778 | 0.002 | 0.00097 | 0.000103 | 0.555113 | 0.00000001 |
| 0.001778 | 0:002 | 0.00097 | 0.000103 | 0.555113 | 0.00000001 |
| 0.001778 | 0.002 | 0.00097 | 0.000103 | 0.555113 | 0.00000001 |
| 0.001778 | 0.002 | 0.00097 | 0.000103 | 0.555113 | 0.00000001 |
| 0.001778 | 0.002 | 0.00097 | 0.000103 | 0.555113 | 0.00000001 |
| 0.001778 | 0.002 | 0.00097 | 0.000103 | 0.555113 | 0.00000001 |
| 0.001778 | 0.002 | 0.00097 | 0.000103 | 0.555113 | 0.00000001 |
| 0.001778 | 0.002 | 0.00097 | 0.000103 | 0.555113 | 0.00000001 |
| 0.001778 | 0.002 | 0.00097 | 0.000103 | 0.555113 | 0.00000001 |
| 0.001778 | 0.002 | - | - | 0.555113 | 0.00000001 |
| 0.001778 | 0.002 | - | - | 0.555113 | 0.00000001 |
| 0.001778 | 0.002 | - | - | - | - |

**Table 6-13:** Different $\sigma_i$ for each legible, illegible and middle class using the selected features.

Table 6-13 shows the best value of different $\sigma_i$ obtained for each legible, illegible and middle style classes classification with an error rate of 0.00994, 0.08337 and 0.00347 for legible/illegible, middle/illegible and legible/middle respectively. The first and second columns show the different $\sigma_i$ values for legible/illegible classification, the third and fourth columns show the different $\sigma_i$ values for middle/illegible classification and the fifth and sixth columns show the different $\sigma_i$ values for legible/middle classification.

| Training set | Test set | % Correct Classification (different $\sigma_i$ ) | % Correct Average |
|---|---|---|---|
| LEGTR16, ILLEGTR16 | LEGTR16 | 99.00% | 99.50% |
| LEGTR16, ILLEGTR16 | ILLEGTR16 | 100.00% | |
| LEGTR16, ILLEGTR16 | LEGTE16 | 93.00% | 78.00% |
| LEGTR16, ILLEGTR16 | ILLEGTE16 | 63.00% | |
| | | Overall | 88.75% |

**Table 6-14:** Classification result using 16 extracted features to discriminate between illegible and legible handwriting using different $\sigma_i$.

| Training set | Test set | % Correct Classification (different $\sigma_i$ ) | % Correct Average |
|---|---|---|---|
| LEGTR15, MiddleTR15 | LEGTR15 | 100.00% | 100.00% |
| LEGTR15, MiddleTR15 | MiddleTR15 | 100.00% | |
| LEGTR15, MiddleTR15 | LEGTE15 | 88.00% | 75.50% |
| LEGTR15, MiddleTR15 | MiddleTE15 | 63.00% | |
| | | Overall | 87.75% |

**Table 6-15:** Classification result using 15 extracted features to discriminate between middle and legible handwriting using different $\sigma_i$.

| Training set | Test set | % Correct Classification (different $\sigma_i$ ) | % Correct Average |
|---|---|---|---|
| MiddleTR13, ILLEGTR13 | MiddleTR13 | 99.00% | 99.50% |
| MiddleTR13, ILLEGTR13 | ILLEGTR13 | 100.00% | |
| MiddleTR13, ILLEGTR13 | MiddleTE13 | 100.00% | 86.00% |
| MiddleTR13, ILLEGTR13 | ILLEGTE13 | 72.00% | |
| | | Overall | 92.50% |

**Table 6-16:** Classification result using 13 extracted features to discriminate between middle and illegible handwriting using different. $\sigma_i$

Tables 6-14, 6-15 and 6-16 show that the overall classification results are 87.50%, 92.75% and 88.75% correct classification when classifying legible/middle, illegible/middle and legible/illegible word images respectively. This can broken down into 100.00%, 99.50% and 99.50% correct classification when the test set is the same as the training set and 75.50%, 86.00% and 78.00% correct classification when the test set is different to the training set.

#### 6.4.1.3 Comparison Between Using The Selected and 36 Extracted Features

Figures 6-14 and 6-15 summarise the results of tables 6-10 to 6-12 (page 6-18) and 6-14 to 6-16 pages (6-19 and 6-20). Figure 6-14 shows the classification result when the training set is the same as test set and figure 6-15 shows the classification result when the training set is different to the test set.

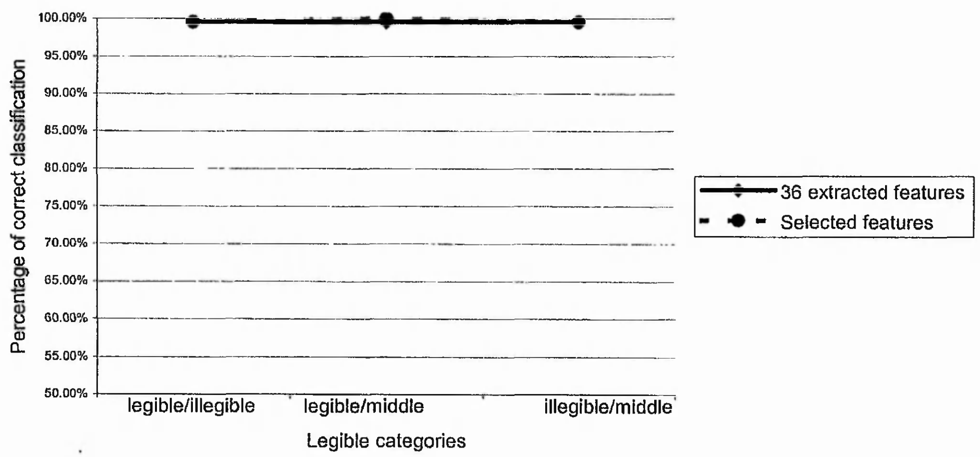



**Figure 6-14:** Comparison between the selected and 36 extracted features using different $\sigma_i$ for seen data.

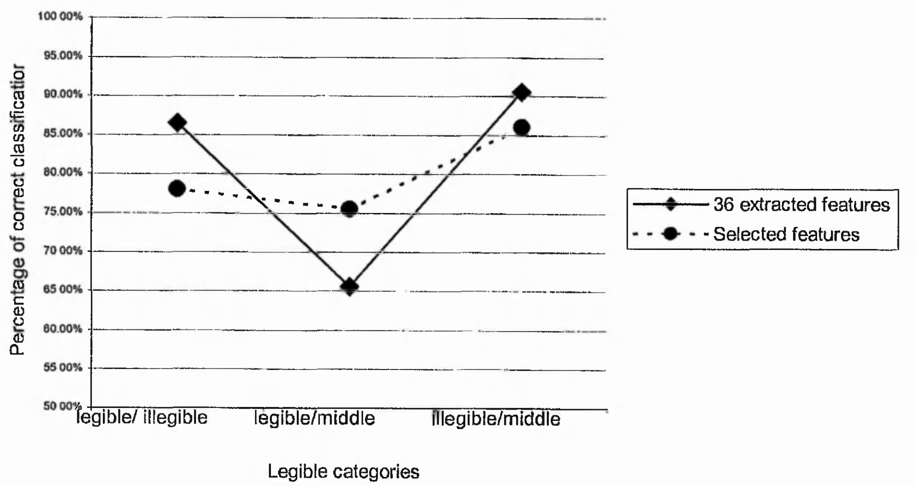



**Figure 6-15:** Comparison between the selected and 36 extracted features using different $\sigma_i$ for unseen data.

Figure 6-14 shows that for classification between legible/illegible, legible/middle and illegible/middle there is virtually no difference using 36 features rather than the selected features when the test set is the same as the training set.

In Figure 6-15 the system achieved an improvement of 8.50%, 4.50% in classification between legible/illegible and illegible/middle and a decrease of 10.00% for legible/middle classification when using 36 features rather than the selected features where the training set is different to the test set.

### 6.4.2 Triple Classification

Tables 6-18, 6-20 shows the experimental results obtained using (i) all 36 extracted features (table 6-18) and (ii) the selected features (table 6-20) for three class (legible, middle and illegible) style classification. The first column shows the samples that were used as the training data set whilst the second column shows the samples that were used as a test set. The third column in table 6-18 shows the correct classification result obtained when using the non-linear (PNN) classification technique using different $\sigma_i$ with 36 extracted features whilst column three in table 6-20 shows the similar result when using the 20 selected features. Columns four, five and six show the misclassification results in each category.

#### 6.4.2.1 Experimental Results and Analysis Using 36 Extracted Features

Table 6-17 shows the best values of different $\sigma_i$ obtained for each legible, illegible and middle classification with an error rate of 0.21840.

| **Legible/Illegible/ Middle $\sigma_i$ in legible class** | **Legible/Illegible/ Middle $\sigma_i$ in Illegible class** | **Legible/Illegible/ Middle $\sigma_i$ in Middle class** |
|---|---|---|
| 0.000889 | 0.000931 | 0.001260 |
| 0.000889 | 0.000931 | 0.001260 |
| 0.000889 | 0.000931 | 0.001260 |
| 0.000889 | 0.000931 | 0.001260 |
| 0.000889 | 0.000931 | 0.001260 |
| 0.000889 | 0.000931 | 0.001260 |
| 0.000889 | 0.000931 | 0.001260 |
| 0.000889 | 0.000931 | 0.001260 |
| 0.000889 | 0.000931 | 0.001260 |
| 0.000889 | 0.000931 | 0.001260 |
| 0.000889 | 0.000931 | 0.001260 |
| 0.000889 | 0.000931 | 0.001260 |
| 0.000889 | 0.000931 | 0.001260 |
| 0.000889 | 0.000931 | 0.001260 |
| 0.000889 | 0.000931 | 0.001260 |
| 0.000889 | 0.000931 | 0.001260 |
| 0.000889 | 0.000931 | 0.001260 |
| 0.000889 | 0.000931 | 0.001260 |
| 0.000889 | 0.000931 | 0.001260 |
| 0.000889 | 0.000931 | 0.001260 |
| 0.000889 | 0.000931 | 0.001260 |
| 0.000889 | 0.000931 | 0.001260 |
| 0.000889 | 0.000931 | 0.001260 |
| 0.000889 | 0.000931 | 0.001260 |
| 0.000889 | 0.000931 | 0.001260 |
| 0.000889 | 0.000931 | 0.001260 |
| 0.000889 | 0.000931 | 0.001260 |
| 0.000889 | 0.000931 | 0.001260 |
| 0.000889 | 0.000931 | 0.001260 |
| 0.000889 | 0.000931 | 0.001260 |
| 0.000889 | 0.000931 | 0.001260 |
| 0.000889 | 0.000931 | 0.001260 |
| 0.000889 | 0.000931 | 0.001260 |
| 0.000889 | 0.000931 | 0.001260 |
| 0.000889 | 0.000931 | 0.001260 |
| 0.000889 | 0.000931 | 0.001260 |

**Table 6-17.** The thirty-six chosen $\sigma_i$ values for each legible, illegible and middle classification using the 36 extracted features.

**%Misclassification words**

| Training files | Test files | %Correct non-linear (PNN) | AS Legible | As Illegible | As Middle | %Correct Average |
|---|---|---|---|---|---|---|
| LEGTR36, ILLEGTR36,MiddleTR36 | LEGTR36 | 100.00% | - | 0 | 0 | 99.33% |
| LEGTR36, ILLEGTR36,MiddleTR36 | ILLEGTR36 | 99.00% | 1.00% | - | 0 | |
| LEGTR36, ILLEGTR20,MiddleTR36 | MiddleTR36 | 99.00% | 0.60% | 0.40% | - | |
| LEGTR36, ILLEGTR36,MiddleTR36 | LEGTE36 | 72.00% | - | 10.00% | 18.00% | 67.33% |
| LEGTR36, ILLEGTR36,MiddleTR36 | ILLEGTE36 | 83.00% | 17.00% | - | 0 | |
| LEGTR36, ILLEGTR36,MiddleTR36 | MiddleTE36 | 47.00% | 51.00% | 2.00% | - | |
| | | | | | Overall | 83.33% |

**Table 6-18:** Classification result using 36 extracted features to discriminate between legible, illegible and middle handwriting using different $\sigma_i$.

Table 6-18 shows that the PNN classifier using different $\sigma_i$ values achieves 67.30% correct classification when the test set is different to the training set and 99.30% correct classification when the test set is the same as the training set. This gives an overall 83.33% correct classification.

### 6.4.2.2 Experimental Results and Analysis Using The Selected Features

Table 6-19 shows the best values of different $\sigma_i$ values obtained for each legible, illegible and middle classification with an error rate of 0.58538.

| Legible/Illegible/ Middle $\sigma_i$ in legible class | Legible/Illegible/ Middle $\sigma_i$ in Illegible class | Legible/Illegible/ Middle $\sigma_i$ in Middle class |
|---|---|---|
| 12.161509 | 29.097356 | 34.525470 |
| 26.934552 | 33.924226 | 27.622357 |
| 14.740815 | 31.123368 | 34.015038 |
| 17.322724 | 35.467820 | 33.008801 |
| 32.337875 | 24.764799 | 32.572552 |
| 37.291618 | 23.355932 | 26.555021 |
| 31.360864 | 27.302276 | 31.065763 |
| 36.515355 | 19.636148 | 30.528229 |
| 32.672058 | 24.540932 | 32.791168 |
| 33.043648 | 23.944269 | 32.801497 |
| 14.272983 | 25.226491 | 29.353932 |
| 31.668817 | 19.013886 | 30.039603 |
| 25.936745 | 24.588337 | 31.032383 |
| 33.357175 | 29.633227 | 23.063293 |
| 33.014600 | 24.055930 | 32.777517 |
| 24.721231 | 31.259538 | 32.906403 |
| 28.609272 | 25.691595 | 33.108511 |
| 23.169188 | 29.953808 | 27.127578 |
| 32.003716 | 49.622118 | 10.927476 |
| 23.795174 | 40.279615 | 36.777781 |

**Table 6-19.** Twenty different $\sigma_i$ for each legible, illegible and middle class using the selected features.

| Training files | Test files | %Correct non-linear (PNN) | %Misclassification words | | | %Correct Average |
|---|---|---|---|---|---|---|
| | | | AS Legible | As Illegible | As Middle | |
| LEGTR20, ILLEGTR20, MiddleTR20 | LEGTR20 | 88.00% | - | 5.00% | 7.00% | 74.33% |
| LEGTR20, ILLEGTR20, MiddleTR20 | ILLEGTR20 | 80.00% | 19.00% | - | 1.00% | |
| LEGTR20, ILLEGTR20, MiddleTR20 | MiddleTR20 | 55.00% | 39.00% | 6.00% | - | |
| LEGTR20, ILLEGTR20, MiddleTR20 | LEGTE20 | 72.00% | - | 8.00% | 20.00% | 44.67% |
| LEGTR20, ILLEGTR20, MiddleTR20 | ILLEGTE20 | 35.00% | 39.00% | - | 26.00% | |
| LEGTR20, ILLEGTR20, MiddleTR20 | MiddleTE20 | 27.00% | 41.00% | 32.00% | - | |
| | | | | | Overall | 59.50% |

Table 6-20: Classification result using 20 selected features to discriminate between legible illegible and middle handwriting using different $\sigma_i$.

These results show that the PNN classifier using different $\sigma_i$ achieves a 44.67% correct classification when the test set is different to the training set and a 74.33% correct classification when the test set is the same as the training set. This gives an overall 59.50% correct classification.

### 6.4.2.3 Comparison Between Using The Selected And 36 Extracted Features

Figures 6-16 and 6-17 show that for classification between legible, illegible and the classification result using all 36 features is better than using the 20 selected features for both seen and unseen data.

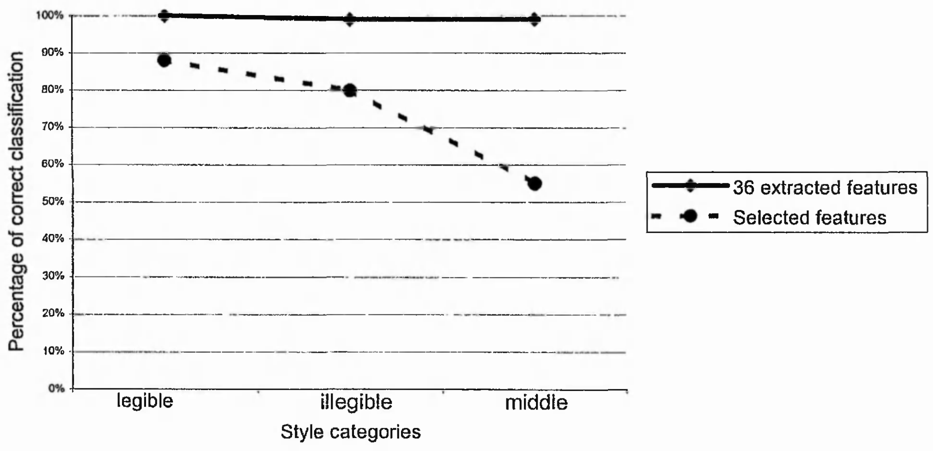



**Figure 6-16:** Comparison between using the selected and 36 extracted features for style classification with seen data.

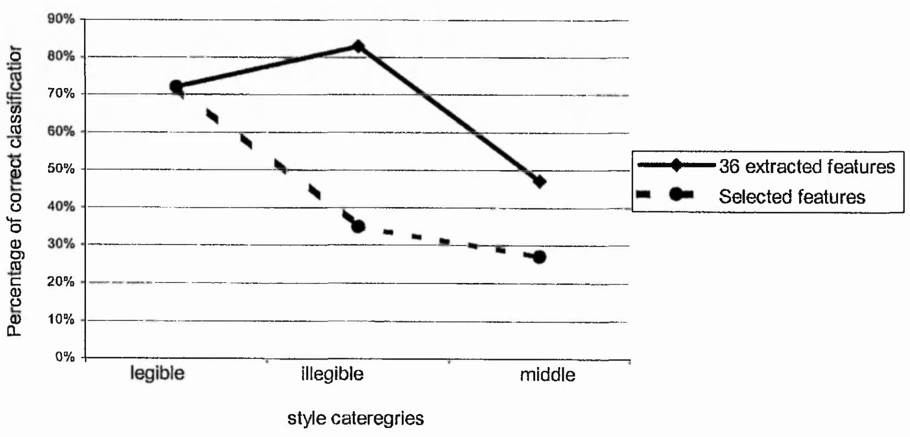



**Figure 6-17:** Comparison between using the selected and 36 extracted features for style classification with unseen data.

## 6.5 Comparison Between Using The Selected And 36 Extracted Features In Triple Classification Using PNN Method

Tables 6-21 and 6-22 summarise the triple classification result achieved in section 6.3 and 6.4.

| Training set is the same as test set | Legible | | Illegible | | Middle | | Overall | |
|---|---|---|---|---|---|---|---|---|
| | Dif $\sigma_i$ | Com $\sigma$ | Dif $\sigma_i$ | Com $\sigma$ | Dif $\sigma_i$ | Com $\sigma$ | Dif $\sigma_i$ | Com $\sigma$ |
| Selected features | 88.00% | **88.00%** | 80.00% | **71.00%** | 55.00% | **39.00%** | 74.33% | **66.00%** |
| 36 extracted features | 100.00% | **100.00%** | 99.00% | **100.00%** | 99.00% | **99.0%** | 99.33% | **99.67** |

**Table 6-21:** Comparison between using different $\sigma_i$ and common $\sigma$ with 20 features and 36 extracted features when the training set is the same as the test set.

| Training set is different with the test set | Legible | | Illegible | | Middle | | Overall | |
|---|---|---|---|---|---|---|---|---|
| | Dif $\sigma_i$ | Com $\sigma$ | Dif $\sigma_i$ | Com $\sigma$ | Dif $\sigma_i$ | Com $\sigma$ | Dif $\sigma_i$ | Com $\sigma$ |
| Selected features | 72.00% | **76.00%** | 35.00% | **42.00%** | 27.00% | **20.00%** | 44.67% | **46.00%** |
| 36 extracted features | 72.00% | **72.00%** | 83.00% | **83.00%** | 47.00% | **47.00%** | 67.33% | **67.33%** |

**Table 6-22:** Comparison between using different $\sigma_i$ and common $\sigma$ with 20 selected and 36 extracted features when the training set is different to the test set.

Details of the above tables will be explained in subsections 6.5.2.1 and 6.5.2.2.

### 6.5.1 36 Extracted Features

The experimental results in tables 6-21 and 6-22 (page 6-30) show that there is no difference between using 36 features with different $\sigma_i$ values and common $\sigma$ when the test set is the same as training set or when the test set is different to the training set.

### 6.5.2 Selected Features

The experimental results in table 6-21 and 6-22 (page 6-30) show that an improvement of 9% and 16% can be achieved by using different $\sigma_i$ rather than the common $\sigma$ for classifying illegible and middle words when the test set is the same as the training set. However there is no improvement between using different $\sigma_i$ or common $\sigma$ when classifying legible word images.

These tables also show a decrease of 4.00% and 7.00% is obtained when using different $\sigma_i$ for classifying legible and illegible words and an improvement of 7.00% is achieved in the classification of middle words when the training set is different to the test set.

## 6.6 Multiple Linear Classification (MDA)

Tables 6-23 to 6-25 show the experimental result obtained using all 36 extracted features to classify between legible/illegible, legible/middle and illegible/middle word images when using the multi-linear discriminant analysis technique. The first column shows the samples that were used as the training data set whilst the second column show the samples that were used as a test set. The third column shows the correct classification result The fourth column shows average of correct classification result when the system was tested with seen or unseen data. The last row then shows the average classification result for all data. The training samples and test samples are the same as those used in the non-linear classification experiment.

| Training set | Test set | % Correct Classification MDA | % Correct Average |
|---|---|---|---|
| LEGTR36, ILLEGTR36 | LEGTR36 | 78.00% | 70.50% |
| LEGTR36, ILLEGTR36 | ILLEGTR36 | 63.00% | |
| LEGTR36, ILLEGTR36 | LEGTE36 | 67.00% | 60.50% |
| LEGTR36, ILLEGTR36 | ILLEGTE36 | 54.00% | |
| | | Overall | 65.50% |

**Table 6-23:** Classification result using 36 features to discriminate between legible and illegible.

| Training set | Test set | % Correct Classification MDA | % Correct Average |
|---|---|---|---|
| LEGTR36, MiddleTR36 | LEGTR36 | 70.00% | 64.00% |
| LEGTR36, MiddleTR36 | MiddleTR36 | 58.00% | |
| LEGTR36, MiddleTR36 | LEGTE36 | 57.00% | 63.50% |
| LEGTR36, MiddleTR36 | MiddleTE36 | 70.00% | |
| | | Overall | 63.75% |

**Table 6-24:** Classification result using 36 features to discriminate between legible and middle.

| Training set | Test set | % Correct Classification MDA | % Correct Average |
|---|---|---|---|
| MiddleTR36, ILLEGTR36 | MiddleTR36 | 66.00% | 64.50% |
| MiddleTR36, ILLEGTR36 | ILLEGTR36 | 63.00% | |
| MiddleTR36, ILLEGTR36 | MiddleTR36 | 56.00% | 57.50% |
| MiddleTR36, ILLEGTR36 | ILLEGTR36 | 59.00% | |
| | | Overall | 61.00% |

**Table 6-25:** Classification result using 36 features to discriminate between middle and illegible.

The overall binary classification using 36 features in the MDA technique is 65.50%, 63.75%, and 61.00% for classification between legible/illegible, legible/middle and illegible/middle words. This can be broken down into 70.50%, 64.00% and 64.50% correct classification when the test set is the same as training set and 60.5%, 63.50% and 57.50% correct classification when training set is different to the test set.

| Training set | Test set | % Correct Classification MDA | % Correct Average |
|---|---|---|---|
| LEGTR16, ILLEGTR16 | LEGTR16 | 83.00% | 71.50% |
| LEGTR16, ILLEGTR16 | ILLEGTR16 | 60.0% | |
| LEGTR16, ILLEGTR16 | LEGTE16 | 74.00% | 57.00% |
| LEGTR16, ILLEGTR16 | ILLEGTE16 | 40.00% | |
| | | Overall | 64.25% |

**Table 6-26:** Classification result using 16 extracted features to discriminate between legible and illegible.

| Training set | Test set | % Correct Classification MDA | % Correct Average |
|---|---|---|---|
| LEGTR15, MiddleTR15 | LEGTR15 | 73.00% | 66.00% |
| LEGTR15, MiddleTR15 | MiddleTR15 | 59.00% | |
| LEGTR15, MiddleTR15 | LEGTE15 | 62.00% | 61.50% |
| LEGTR15, MiddleTR15 | MiddleTE15 | 61.00% | |
| | | Overall | 63.75% |

**Table 6-27:** Classification result using 15 extracted features to discriminate between legible and middle.

| Training set | Test set | % Correct Classification MDA | % Correct Average |
|---|---|---|---|
| MiddleTR13,ILLEGTR13 | MiddleTR13 | 66.00% | 59.00% |
| MiddleTR13,ILLEGTR13 | ILLEGTR13 | 52.00% | |
| MiddleTR13,ILLEGTR13 | MiddleTR13 | 67.00% | 59.00% |
| MiddleTR13,ILLEGTR13 | ILLEGTR13 | 51.00% | |
| | | Overall | 59.00% |

**Table 6-28:** Classification result using 13 extracted features to discriminate between middle and illegible.

Tables 6-26 to 6-28 show the experimental result obtained when using the selected features using (MDA) technique detailed in table 3-2 chapter 3. The overall binary classification when using selected features in the MDA technique is 64.25%, 63.75% and 59.00% for classification between legible/illegible, legible/middle and illegible/middle words. This can broken down into 71.50%, 66.00% and 59.00% correct classification when the test set is the same as training set and 57.00%, 61.50% and 59.00% correct classification when the training set is different with the test set.

## 6.7 Comparison Between Using the Linear and Non-linear Method for Binary Classification

Tables 6-28 and 6-30 summarise the experimental result obtained when using (i) selected features or (ii) all 36 extracted features using PNN technique with common $\sigma$ or different $\sigma_i$ and (iii) MDA technique.

| Training set is the same as test set | Legible/Illegible | | | Illegible/Middle | | | Middle/Legible | | | Overall | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | Dif $\sigma_i$ | Com $\sigma$ | MDA | Dif $\sigma_i$ | Com $\sigma$ | MDA | Dif $\sigma_i$ | Com $\sigma$ | MDA | Dif $\sigma_i$ | Com $\sigma$ | MDA |
| Selected features | 99.50% | **99.50%** | 71.50% | 99.50% | **99.50%** | 59.00% | 100.00% | **100.00 %** | 66.00% | 99.67% | **99.67%** | 68.83% |
| 36 extracted features | 99.50% | **99.50%** | 70.50% | 99.50% | **99.50%** | 64.50% | 99.50% | **99.50%** | 64.00% | 99.5% | **99.50%** | 66.33% |

**Table 6-29:** Comparison between the classification results when (i) PNN using different $\sigma_i$, (ii) PNN using common $\sigma$ and (iii) MDA techniques when the training set is the same as the test.

| Training set is different with the test set | Legible/Illegible | | | Illegible/Middle | | | Middle/Legible | | | Overall | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | Dif $\sigma_i$ | Com $\sigma$ | MDA | Dif $\sigma_i$ | Com $\sigma$ | MDA | Dif $\sigma_i$ | Com $\sigma$ | MDA | Dif $\sigma_i$ | Com $\sigma$ | MDA |
| Selected features | 78.00% | **78.00%** | 57.00% | 86.00% | **80.00%** | 59.00% | 75.50% | **70.5%** | 61.50% | 79.83% | **76.17%** | 59.17% |
| 36 extracted features | 86.50% | **79.50%** | 60.50% | 90.5% | **76.00%** | 57.50% | 65.50% | **65.50%** | 63.50% | 80.83% | **73.67%** | 60.50% |

**Table 6-30**: Comparison between the classification results when (i) PNN using different $\sigma_i$, (ii) PNN using common $\sigma$ and (iii) MDA techniques when the training set is different to the test set

### 6.7.1 36 Extracted Features

The experimental result given in tables 6-29 and 6-30 shows that the PNN technique achieved an improvement of 26.00%, 2.00% and 33% using different $\sigma_i$ and an improvement of 19.00%, 2.00% and 18.50% using common $\sigma$ when compared to the MDA technique for classification between legible/illegible, legible/middle and illegible/middle words respectively where the test set is different to the training set. In the case where the training set is the same as the test set the PNN technique achieved an improvement of 29.00%, 35.50% and 35.00% using different $\sigma_i$ and an improvement of 29.00%, 35.50% and 35.00% using common $\sigma$ compared to the MDA technique for classification between legible/illegible, legible/middle and illegible/middle words respectively.

The experimental result given in table 6-29 show that when the training set is the same as the test set there is no difference in classification rate between using different $\sigma_i$ values and common $\sigma$ value. However, table 6-30 shows that whilst using different $\sigma_i$ rather than common $\sigma$ has no affect on the classification between legible/middle it does give an improvement of 7.00% and 14.50% for classification between legible/illegible, illegible/middle when the test set is different to the training set.

### 6.7.2 Selected Features

The experimental result tables in tables 6-29 and 6-30 shows that PNN technique achieved an improvement of 28.00%, 44.00% and 40.50% using different $\sigma_i$ and an improvement of 28.00%, 44.00% and 40.50% using common $\sigma$ when compared to the MDA technique for classification between legible/illegible, legible/middle and illegible/middle words respectively where the test set is the same as training set. In the case where the training set is different to the test set PNN technique achieved an improvement of 21.00%, 14.00% and 27.00% using different $\sigma_i$ and an improvement of 21.00%, 9.00% and 21.00% using common $\sigma$ achieved compared to the MDA technique for classification between legible/illegible/middle words respectively.

## 6.8 Conclusion

In this chapter legibility of handwriting based on an existing recogniser has been defined. Then two methods for the legibility classification of the word images are described (MDA and PNN) and a comparison between these two methods is presented. Experimental result using MDA and PNN techniques using different $\sigma_i$ show that in the case of legibility/illegible and illegible/middle the PNN technique using different $\sigma_i$ gives the superior result compared to using the PNN with common $\sigma$ and the MDA technique using 36 features. However, in the case of middle/legible classification the PNN technique using common $\sigma$ with selected features gives a better classification result.

# 7. CONCLUSION AND FUTURE WORK

In order to help improve recognition accuracy a lot of research has been directed towards dealing with the variability of handwriting prior to recognition. This research is another attempt to address the problem associated with the variability of human handwriting. Novel approaches of using MDA and PNN systems to predict the case and legibility of handwriting prior to recognition are used for this purpose. As the ability of a machine-based recogniser differs significantly from that of a human being a novel definition of legibility based on the recognition system is constructed. The research was to investigate the potential for using handwriting case classification (upper, lower and mixed case words) and legibility classification of handwriting (as determined by the existing recogniser) to help improve CSR accuracy.

In this research we show that a pre-classification of words into upper, lower and mixed case could provide a useful means of reducing ambiguity. By successfully classifying the case of words prior to recognition the size of the lexicon used for any individual word recognition could be reduced which in turn should improve the recognition results. We also show that a classification of handwriting style on the basis of recogniser specific legibility could be successfully used to select style specific recognisers prior to recognition. Such a system consumes less memory and computation resources and exhibits less confusion errors.

Two stages dominated this research;

1. The initial stage of the research concentrated on feature extraction. The idea is to extract information from the handwriting input, not in order to identify the writer, but to find information about the style of characters or words. Thirty-six features were introduced and an automatic feature evaluation method based on MDA was proposed and verified. The

effectiveness of each feature in a classification between each pair of class (lower/upper, lower/mixed, upper/mixed, legible/illegible, legible/middle and illegible/middle) and all three classes (upper/lower/mixed and legible/illegible/middle) was examined using Multiple Discriminant Analysis.

Experimental results show that some of the features have a more significant influence on classification results than the others (see table 4-2 page 4-16 and table 4-4 page 4-21). However experiments also show all the features used in this research play some role and are deemed necessary for successful classification. Indeed a significant reduction of feature vectors leads to a much less effective classification.

2. The second stage of research investigated techniques for style classification of handwriting. This work has introduced a novel handwriting legibility classification system that can be used to predict the recognition performance of a recogniser for a given handwriting style in order to choose the best recogniser.

Two methods, Multiple Discriminant Analysis and a Probability Neural Network were used in the classification phase and a comparison between the two methods was presented for case and legibility classification in chapters 5 and 6. The MDA technique was used to create a nearest-mean classifier using the Euclidean distance to find the nearest neighbours whilst the PNN technique used a Bayes decision rule and a Parzen model to estimate the class conditional density. With the PNN method a classifier was designed using $(n-1)$ samples and evaluated on the one remaining sample; this is repeated $n$ times with different training sets of size $(n-1)$ (leave-one-out method) to estimate the error rate.

## 7.1 Achievements in Case Classification of Handwriting

The results show that for upper/lower word case classification using the selected features with the PNN technique (common $\sigma$) gave the best classification result (100%) when compared to the other techniques (PNN with different $\sigma_i$ and MDA) on unseen data.

The experimental results also show that the PNN technique using different $\sigma_i$ values gives the best result in the case of upper/mixed and mixed/lower classification when using 36 features. The classification results were 88.0% and 81.0% respectively.

For triple classification (upper/lower/mixed) using the PNN with selected features again gave a slightly better classification result than using 36 extracted features. The best classification technique was when using the PNN with different $\sigma_i$. The overall classification results were 73% using selected features and 71.67% using the 36 extracted features respectively on unseen data.

These case classification results are promising especially when it is compared to previous research in this area. As mentioned in chapter 2, Ho and Nagy [HO01] present results for identifying lowercase and uppercase characters, digits and punctuations in a text document. However, their work is on optical character in comparison with our research, which operates on cursive script word images. This is a much more difficult problem as the variability of cursive script is far greater than anything encountered in printed writing.

## 7.2 Achievements in Handwriting Legibility Classification

The experimental results show that using 36 features in a PNN system with different $\sigma_i$ gave better results for legible/illegible and illegible/middle classification than using selected features. The results are 86.50% and 90.5% respectively on unseen data. The best result achieved for middle/legible classification was when using the PNN technique with the selected features. The classification result was 75.50%. However overall the best single classifier for binary classification of legibility was using the PNN with different $\sigma_i$ and 36 features. The overall classification result is 80.83%.

For triple classification the best classification technique was the PNN with 36 features. In this case the classification result was 67.33%. There is no difference between PNN using common $\sigma$ or different $\sigma_i$.

As the PNN in classification between two classes gives superior results in comparison to the MDA, in this research we use PNN for triple classification and no experiments were carried out for the triple classification with the MDA technique. Experimental results show that those words, which were correctly classified using the MDA technique, were equally correctly classified using PNN. However, those words, which were misclassified or closely classified PNN, were correctly classified using MDA.(Appendix B)

The methods presented here have already been published in the proceeding of four international conferences of high standing denoting that they represent a significant contribution to the knowledge of the scientific community in the area of style classification of cursive handwriting. The result of this work contains two key contributions. Firstly, the work has demonstrated that the pre-processing of cursive handwriting to upper, lower and mixed case word images can be achieved to a workable level of accuracy. Secondly, the pre-classification of unseen cursive word images into legible, illegible and middle on the basis of an existing recogniser's performance on the training set has been demonstrated.

## 7.3 Summary

Providing a means of pre-classifying word images into upper, lower and mixed case is expected to provide a significant contribution as currently most of the reported algorithms simply assume this pre-classification. We have shown that the method presented here is capable of classifying the word into upper, lower and mixed case with high accuracy. The accuracy for lower/upper, upper/mixed and lower/mixed classification are 100%, 88% and 81% respectively (see chapter 5). This could be used as a means of limiting the size of the template database for word recognition therefore the recogniser spends less time in searching space, consumes less memory and improves the accuracy. Consequently the costs of computational expenses are significantly reduced.

In practice, as the results of any misclassification turn to reduce the recognition result, therefore, the accuracy of any classification should be high in a pre-processing stage. The PNN technique provides 86.5% (legible/illegible), 90.5% (middle/illegible) correct classification (see chapter 6). Although further word level features are needed to further improve classification between legible/ middle, this result is also significant. In practice by using this technique we can distinguish between illegible and an other words. Another advantage of using the PNN method at this stage is that we can gain confidence level before any recognition that depends on the applications. In other words the idea of introducing rejection categories will be considered with a view to providing a confidence measure for legibility classification. By using confidence level in the classification phase the system let us know which recogniser is best for the specific word.

The remaining sections provide suggestions and discussion, which concentrate on how to expand the developed method to improve on partially working and non-working areas and how the result can be used to improve the recognition performance.

## 7.4 Future Works

This section presents the possible areas of future investigation that could link this work with other projects in the future. As the size and quality of writing is important in these experiments, some of the features are not extracted correctly, resulting in misclassification. It is therefore suggested that further examination of the selected features should be considered. One possible candidate is fractals. Fractal features may provide useful information to discriminate between legible/illegible/middle handwriting word images. These features have been useful for classifying the regularity in handwriting as well as size of writing [BOULETREAU97].

The Parzen model, used for density estimation in the PNN system, has the same number of kernels as the number of data points. This leads to models that can be slow to evaluate for new input vectors especially when the number of training data points is very large. One way to tackle this problem is to use a clustering technique such as fuzzy clustering to reduce the number of data points prior to PNN. The centre of each cluster can be used as a centre for each kernel thus greatly increasing the classification speed.

Faced with significant style variation of handwriting it is more likely that style-specific classifiers yield higher classification accuracy than the generalised classifiers. Therefore, the next stage of our work would be to use the pre-classifier to route a given data sample to a recogniser which is deemed more suitable to the style of the sample. The work so far has concentrated on a small subset of style classification. The result of our initial experiments in applying the described techniques to determine a writer style has been encouraging.

Further investigation to determine how effectively we can identify a writer will be needed. It is a fact that intra-writer style variation is also a problem [JEDRZEJEWSKI97]. This can lead to significant user frustration such as affects today's on-line applications (PDAs). It would be interesting to see whether there is any scope in treating intra-writer style variation in a similar way.

These classification methods can also be applied for identifying the symbol types such as digit, punctuation and lower, upper letters for further work [HO01]. For example separation of digits and uppercase, lowercase characters or words is an important task in document layout. This method could be very useful in the field of writer and signature identification. Using the methods presented here it may be possible to determine the characteristics of each writer using the most efficient features in each writer's handwriting.

## REFERENCES

[ASHTOSH97] Ashutosh Malaviya and Lilian Peters, "Fuzzy feature description of handwriting patterns", Pattern recognition, Vol. 30, No. 10, pp. 1591-1604, 1997.

[BARID94] H. S. Baird and G. Nagy.,"A self-correcting 100-font classifier", In L. Vincent and T. Pavlidis, editors, Document Recognition, Proceedings of the SPIE, volum 2181, pages 106-115, 1994.

[BERRIN94] Berrin A. Yanikoglu and Peter A. Sandon, " Recognition off-line cursive handwriting", IEEE, 1994.

[BERRIN98] Berrin A. Yanikoglu and Peter A. Sandon, " Segmentation of off-line cursive handwriting using linear programming", Pattern recognition, Vol. 31, No. 12, pp. 1825-1833, 1998.

[BOZINOVIC89] R.M. Bozinovic and S. N, Srihari , "Off-line cursive script word recognition", IEEE Trans. PAMI, Vol. 11, No.1, pp. 68-83, 1989.

[BOULETREAU97] V. Bouletreau, N. Vincent, R. Sabourin and H.Emptoz," Synthetic Parameters for Handwriting Classification", Forth international conference on document analysis and recognition (Icdar'97), pp. 102-106, 1997.

[BROWN80]M. K. Brown and S. Ganapathy, "Cursive script recognition", proc. Int'I conf. Cybernetic and soc. Pp. 47-51,1980.

[BRITTO00] R. M. Bozinovic and S.N. Srihari, "Off-line cursive script recognition", IEEE Trans. PAMI, Vol. 11, No. 1, pp. 68-83, 1989.

[BUNKE94] H. Bunke, M. Roth, E.G. Schukat-Talamazzini, "Off-line recognition of cursive script produced by a cooperative writer", Procceding s international conference on pattern recognition", pp. 383-386, 1994.

[CAMASTRA01] F.Camastra, A. Vinciarelli, " Cursive character recognition by learning vector quantization", pattern recognition letters, vol. 22, No.6-7, pp. 625-629, 2001.

[CASEY95] G. Casey, E. Lecolinet, "Strategies in character segmentation: A survey", Proceeding 3rd international conference on document analysis and recognition, vol.1, pp. 1028-1033, 1995.

[CHA01] Sung-Hyuk Cha, Sargur N.Srihari, " Apriori Algorithm for sub-category classification Analysis of Handwriting", Sixth international conference on document analysis and recognition (Icdar'01), pp. 1022-1025, 2001.

[CHAN98] Kam-fai Chan, Dit-Yan Yeung, "Elastic Structural Matching for on-line handwritten Alphanumeric Character recognition", Proceeding of 14th international conference on pattern recognition (ICPR'98), vol. 1, pp. 1508-1511, 1998.

[CHAUDHURI98] B. B. Chaudhuri and U. Garain, "Detection of Italic, Bold and All-Capital words in Document Images", proceeding of 14th international conference on pattern recognition (ICPR'98), vol. 1, pp. 610-612, 1998.

[CHEN92] Chen M. Y. et al, "Off-line handwriting word recognition using hidden markov model", proceeding of the 5th USPS Advanced technology conference, pp. 563-577, 1992.

[CHEN93] Chen M., Y. Kundu, A.," An alternative to variable duration HMM in handwriting word recognition", proceeding of the third international workshop on frontiers in handwriting recognition (IWFHR-3), pp. 82-91, 1993.

[CHIEN98] C H Chien and J K Aggarwal.Model, "Construction and Shape Recognition From Occluding Contours", IEEE transaction on pattern analysis and machine intelligence, vol.11, no. 4, pp. 372-389, 1998.

[COATES01] Allison L. Coates, Henry S. Baird and Richard J. Fateman, "Pessimal print: A reverse turing test", Sixth international conference on document analysis and recognition (Icdar'01), pp 1154-1158, 2001.

[CONNELL02] S. D. Connell, A. K. Jain, " Writer adaptation for online handwriting recognition", IEEE transaction on pattern analysis and machine intelligence, vol. 24, No. 3, pp. 329-347, 2002.

[CRETTEZ95] Jean-pierre Crettez, A set of handwriting families: style recognition", Third International conference on document analysis and recognition (Icdar'95), pp. 489-494,1995.

[DING99] Y. Ding, F. Kimura, Y. Miyake, M. Shridhar, " Evaluation and improvement of slant estimation for handwriting words", Fifth international conference on document analysis and recognition (Icdar'99), pp. 753-756, 1999.

[DING00] Y. Ding, F. Kimura, Y. Miyake and M. Shridhar, "Accuracy improvement of slant estimation for hand-written words", Proceeding of 15th international conference on pattern recognition (ICPR'00), vol. 4, pp. 527-530, 2000.

[DODEL95] J. P. Dodel and R. Shinghal, "Symbolic/Neural recognition of cursive amount on bank cheques", Proc. Third Int'l conf. Document analysis and recognition (Icdar'95), pp. 15-18,1995.

[DONALD90] Donald F. Specht, "Probabilistic Neural Networks and the polynomial adaline as comlementary techniques for classification", IEEE transaction on networks, vol. 1, no.1, 1990.

[EBADIAN99a] M. Ebadian Dehkordi, N. Sherkat and R.J. Whitrow, "Classification of Off-line Handwriting Words into Upper and Lower Case", Document Image Processing and Multimedia, IEE, London, March, 1999.

[EBADIAN99b] M. Ebadian Dehkordi, N. Sherkat and R. J. Whitrow, "A Principal Component Approach to Classification of Handwriting Words", Fifth International Conference on Document Analysis and Recognition (ICDAR'99), 781-784, India, September 1999.

[EBADIAN00] M. Ebadian Dehkordi, N. Sherkat and Tony Allen, "Case Classification of Off-line Handwritten Words Prior to Recognition", Fourth International Conference on Document Analysis and System (DAS'00), 325-334, Rio de Janeiro, December 2000.

[EBADIAN01] M. Ebadian Dehkordi, N. Sherkat and Tony Allen, "Prediction of Handwriting Legibility", Sixth International Conference on Document Analysis and Recognition (ICDAR'01), Seattle, September 2001.

[FAVATA01] J. T. Favata, " Offline general handwritten word recognition using an approximate BEAM matching algorithm", IEEE transaction on pattern analysis and machine intelligence, vol. 23, No. 9, pp. 1009-1021, 2001.

[FREEMAN61] H. FREEMAN, "On the encoding of arbitrary geometric configuration", IRE transactions on electronic computers, EC-10 (2): 260-268, 1961.

[FUKUNAGA89] Fukunaga K, Hayes R. "Estimation of classifier performance". Transaction on pattern analysis and machine intelligence 1989; 11:1087-1097.

[GILLOUX94] M. Gilloux, "Writer adaptation for handwritten word recognition using hidden markov models", Twelfth international conference on pattern recognition, Jerusalem, pp. 135-139, 1994.

[GUILLEVIC94] D. Guillevic and C. Y. Suen, "Cursive script recognition: A sentence level recognition scheme", Proceeding of 4th international workshop on frontiers in handwriting recognition (IWFHR'94), pp. 216-223, Dec. 1994.

[GARAIN99] U. Garain, B.B. Chaudhori, "Extraction of type style based meta-information from image documents", Third international conference on document analysis and recognition (Icdar'99), pp. 341-344, 1999.

[GONZALEZ93] R.C. Gonzalez and R. E. Wood. "Digital image processing", Addison Wesley, 1993.

[HAMANAKA00] M. Hamanaka and Yamada, "On-line character recognition adaptively controlled by handwriting quality", proceeding of 7th international workshop on frontiers in handwriting recognition (IWFE'2000), pp. 33-42, Netherlands, 2000.

[HEIJDEN95] Ferdinand Van Der Heijden, " Image based measurement system object recognition and parameter estimation", Wiley, 1995.

[HO01] Tin Kam Ho, George Nagy, " Exploration of contextual constraints for character pre-classification", Sixth international conference on document analysis and recognition (Icdar'01), pp. 450-454, 2001.

[HU99] J. Hu, D. Yu, H. Yan, "Construction of partitioning paths for touching handwritten characters", Pattern recognition letters, vol. 20, no. 3. pp. 293-303, 1999.

[HUANG93]Y. S. Huang and C. Y. Suen, "The Behavior-knowledge space method for combination of multiple classifier", proc. IEEE conf. Computer vision patteren recognition, pp. 347-352, 1993.

[HULL91] J. J. Hull, T. K. Ho and S. N. Srihari, "Word recognition with multilevel contextual knowledge", proc. First int'l conf. Document analysis and recognition (Icdar'91), pp.905-915, 1991.

[IMPEDOVO91] S. Impedow, L. Ottaviano and S. Occhinero, "Optical character recognition: A survey", int' I J. pattern recognition and artificial intelligence, vol. 5, No. 2, pp. 1-24, 1991.

[JAIN00] Anil K. Jain, Robert P. W. Duin and Jianchang Mao, " Statistical pattern recognition: A review", IEEE Transaction on pattern analysis and machine intelligence, vol.22, No.1, January 2000.

[JE91] Jackson Je, " A user's guide to principal components", New York: Wiley, 1991.

[JEDRZEJEWSKI97] Marcin S Jedrzejewski, "Automatic characterisation of handwriting style", MPhil thesis, Department of computing Nottingham Trent University, 1997.

[JUNG99] M. Jung, Y. Shin and S. N. Srihari, " Mltifont classification using typographical attributes", Fifth internatinal conference on document analysis and recognition (Icdar'99), pp. 353-356, 1999.

[JUNG01] m. Jung, Y. C. Shin and S. N. Srihari, " Multifont classification using typographical attributes", Sixth international conference on document analysis and recognition (Icdar'01), pp. 353-356, 2001.

[KAHAN87] S.Kahan, T. Pavlidis and H. S. Baird, "On the recognition of printed characters of any font and size", IEEE transactions on pattern analysis and machine intelligence, vol. 9, no. 2, pp. 274-288, 1987.

[KASSEL] Kassel R. H., "A comparison of approaches to on-line handwritten character recognition", Doctoral Dissertation, Department of electronical engineering and computer science, Massachusetts institute of technology, June 1995.

[KAVALLIERATOU00] E. Kavallieratou, N. Fakotakis and G. Kokkinakis, "A slant removal algorithm", pattern recognition, vol. 33, No. 7, pp.1261-1262, July, 2000.

[KIM97] G. Kim and V. Govindaraju, "A lexicon driven approach to handwritten word recognition for real-time applications", IEEE Trans. PAMI, vol.19, No. 4, pp. 366-379, April 1997.

[KIMURA93] F. Kimura, M. Shridhar and Z. Chen, "Improvements of a lexicon directed algorithm for recognition of unconstrained handwritten words", Second international conference on document analysis and recognition (Icdar'93), pp. 18-22, Oct 1993.

[KUHNKE95] K. Kuhnke, L. Simoncini and Z. Kovacs-V, "A system for machine writer and handwritten character distraction", In proc. Third international conference on document analysis and recognition (Icdar'95), pp. 811-814,1995.

[LEEDHAM94] G. Leedham, "Historical perspective of handwriting recognition system", IEE colloquium on handwriting and pen-based input, (Digest No. 1994/065). london, uk, pp. 1/1-3, march 1994.

[LEROUX91] M. Leroux, J. C. Salome and J. Badard, "Recognition of cursive words in a small lexicons", Proceeding International conference document analysis and recognition, pp.774-782, 1991.

[LI93] Li and N. s. Hall, " Corner detection and shape classification of on-line handprinted Kanji strokes". Patter recognition, vol. 26, No. 9, pp. 1315-1334, 1993.

[LONCARIC98] S. Loncaric, " A survey of shape analysis techniques", pattern recognition, vol. 31, No. 8, pp. 983-1001, 1998.

[MADHVANATH01] S. Madhvanath,V. Govindaraju," The role of holistic paradigms in handwritten word recogniton", IEEE transaction on pattern analysis and machine intelligence, vol. 23, No. 2, pp. 149-165, 2001.

[MIURA97] K. T. Miura, R. Sato, S. Mori, " A method of extracting curvature features and its application to handwritten character recognition", Forth international conference on document analysis and recognition (Icdar'97), pp. 450-454, 1997.

[MORI91] S. Mori, C. Y. Suen and K. Yamamoto, "Historical review of OCR research and development", Proc. IEEE, vol. 80, No. 7, pp. 1029-1058, 1991.

[NAGY00] G. Nagy, " Twenty years of document image analysis in PAMI", IEEE transaction on pattern analysis and machine intelligence, vol. 22, No. 1, pp. 38-62, 2000.

[NICCHIOTTI97] G. Nicchiotti and . Scagliola, "Generalised projections: a tool for cursive handwriting normalisation", Fifth international conference on document analysis ad recognition (Icdar'97), pp. 729-732, sep. 1997.

[PAL01] U. Pal, B.B. Chaudhuri, "Machine-printed and Handwritten text line identification", Pattern recognition letters, vol. 22, No. 3-4, 2001.

[PARZEN62] Parzen, E., "On estimation of a probability density function and mode", Annuals of mathematical statistics, 33:1065-1076, 1962.

[PLAMONDON00] Rejean Plamondon, S. N. Srihari, " On-line and off-line handwriting recognition: A comprehensive survey", IEEE transaction on pattern analysis and machine intelligence, vol. 22, No. 1, 2000.

[POWALKA93] Powalka R. K., Sherkat N., Whitrow R.J. " A toolbox for recognition of varied handwritten script", First European conference on postal technology JET POSTE 93, Nantes, France, pp. 140-147, June 1993.

[POWALKA95] Powalka R. K., Sherkat N., Whitrow R.J. "Recognizer characterisation for combining handwriting recognition", Third international conference on document analysis and recognition (Icdar'95), vo1. 1, pp 68-73, 1995.

[POWALKA96] Powalka R. K., Sherkat N., Whitrow R.J. "Multiple recognition combination topologies", Handwriting and drawing research:... and Applied Issues. pp 329-342, IOS press, 1996.

[ROSEMARY97] A. Rosemary Tate, "Statistical Pattern Recognition for the Analysis Biomedical Magnetic resonance spectra", Journal of Magnetic Resonance Analysis 1997.

[RIPLEY97] B. D. Ripley, "Patern recognition and neural networks", Cambrige, 1997.

[SCHIOLER92] Schioler, H., and Hartmann, U. "Mapping Neural Network Derived from the Parzen window Estimator", Neural Networks, 5(6): pp. 903-909, 1992.

[SCHOMAKER94] L.Schomaker, G. Aabbink and S. Selen, " Writer and writing-style classification in recognition of on-line handwriting", Proceeding of the European workshop on handwriting analysis and recognition, The institute of electrical engineers, (ISSN 0963-3308), LONDON, 1994.

[SHERKAT99] N. Sherkat, T. J. Allen, "Whole word recognition in facsimile images", Fifth international conference on document analysis and recognition (icdar'99), pp. 547-550, 1999.

[SIMNER96] M. I. Simner, C.G. Leedham, A. J. M. Tgomassen(EDS), "Cursive script recognition: A survey", Handwriting and Drawing research:Basic and Applied Issues, pp. 267-284, 1996.

[SIMONCINI95] L. Simoncini and Zs. M. Kovacs-V, "A system for reading USA census'90 Hand-written fields", Third international conference on document analysis and recognition (Icdar'95), vol. II, pp. 86-91, 1995.

[SPECHT91] Specht, Donald F., and Shapiro, Philip D. "Generalization Accuracy of probabilistic Neural Networks compared with back-propagation networks." Lockheed Missiles & space co., Inc. Independent research project RDD 360, I-887-I-892, 1991.

[SRIHARI01] Sargur N. Srihari, Sung_Hyuk Cha, Hina Arora, Sangjik Lee," Individuality of handwriting: A validation study", Sixth international conference on document analysis and recognition (Icdar'01), pp 1195-1204.

[SRIHARI] Sargur N. Srihari, J. J. Hull, "Character recognition", Encyclopaedia of artificial intelligence. S. C. Shapiro, ed. Second ed, pp. 138-150.

[SRIKANTAN96] G. Srikantan, S. W. Lam, and S. N. Srihari, "Gradient- Based contour encoding for character recognition", pattern recognition, vol.29, no. 7, pp. 1147-1160, 1996.

[SUEN92] C. Y. Sune, C. Nadal, R. Legault, T. A Mai and Lam, " Computer recognition of constrained handwritten numerals", Special issue of proceeding of the IECC, vol.80, No. 7, pp.1162-1180, 1992.

[SUEN93] C. Y. Sune, R. Legault, C. Nadal, M. Cheriet and L. Lam, "Building a new generation of handwriting recognition systems", Pattern recognition letters, vol.14, no.4, pp. 303-315, 1993.

[TAPPER84] Tapper C.C., Adaptation on-line handwriting recognition", IEEE 7th international conference. on pattern recognition, pp. 1004-1007, 1984.

[TRIER96] O. D. Trier, A. K. Jain and T. Taxt, "Feature extraction method for character recognition-A survey", Pattern recognition, vol. 29, no. 4, pp.641-662, 1996.

[TURING50] A. Turing, " Computing machinery and intelligence", mind, vol. 59(236), pp. 433-460, 1950.

[UCHIDA01] Seiichi Uchida, Eiji Taira and Hiroaki Sakoe, "Nonuniform slant correction using dynamic programming", Sixth international conference on document analysis and recognition (Icdar'01), pp. 434-438, 2001.

[UNDERWOOD82] G. Underwood and K. Bargh, "Word shape, Orthograhic regularity and contextual interactions in a reading task", cognition, vol.12, pp.197-209, 1982.

[VERMA98] B. Verma, M. Blumenstein and S. Kulkarni, "Recent achivement in off-line handwriting recognition systems", International conference on computational intelligence multimedia application. Australia, pp.27-33, 1998.

[VINCENT01] N. Vincent and T. Freche, " Gray level use in a Handwriting Fractal approach and Morphological Properties quantification", Sixth international conference on document analysis and recognition (Icdar'01), pp. 307-311, 2001.

[VINCIARELLI00] A.Vinciarelli, "A survey on off-line cursive script recognition", IDIAP research report, Switzerland, 2000.

[VUURPIJL96] L.Vuurpijl, OL.Schomaker, "Coarse writing-style clustering based on simple stroke-related features", Proc 5th international workshop frontiers in handwriting recognition, pp 29-36, September 1996.

[WAARD95] W. P. De Waard, "An optimised minimal edit distance for handwriting word recognition", pattern recognition letters, vol.16, no.10, pp.1091-1096, 1995.

[WANG99] J. Wang, H. Yan, "Mending broken handwriting with a macrostructure analysis method to improve recognition", pattern recognition letters, vol. 20, no. 8, pp. 855- 864, 1999.

[WEB62] Andrew Webb, "Statistical pattern recognition", Arnold, 1999.

[YAMAMOTO84] K. Yamamoto, S. Mori, and M. Yasuda, "Research on machine recognition of handprinted characters", IEEE Trans. Pattern analysis and machine intelligence, vol.6, no.4, pp.386-405,1984.

[ZHOA95] Shila X. Zhoa and Sargur N. Srihari, " Word recognition using a lexicon costrained by first/last character decision", SPIE, vol. 2422, pp. 98-104, 1995.

[ZHU99] Yong Zhu, Tieniv Tan and Yunhong Wang, " Font recognition based on global texture analysis", Fifth international conference on document analysis and recognition (Icdar'99), pp. 349-352, 1999.

# APPENDIX A. Some samples from the database

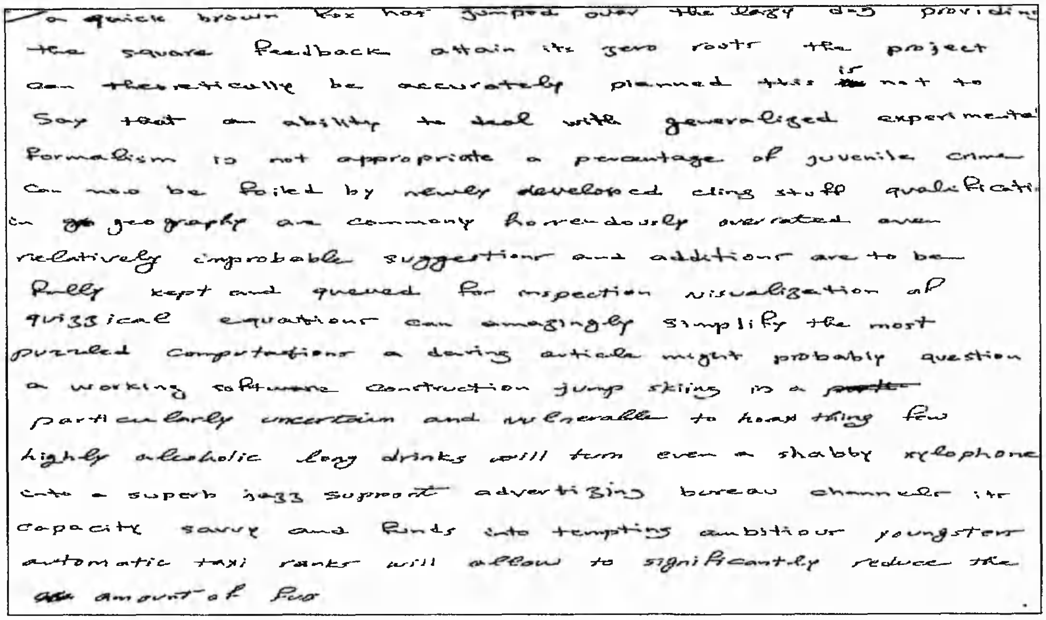
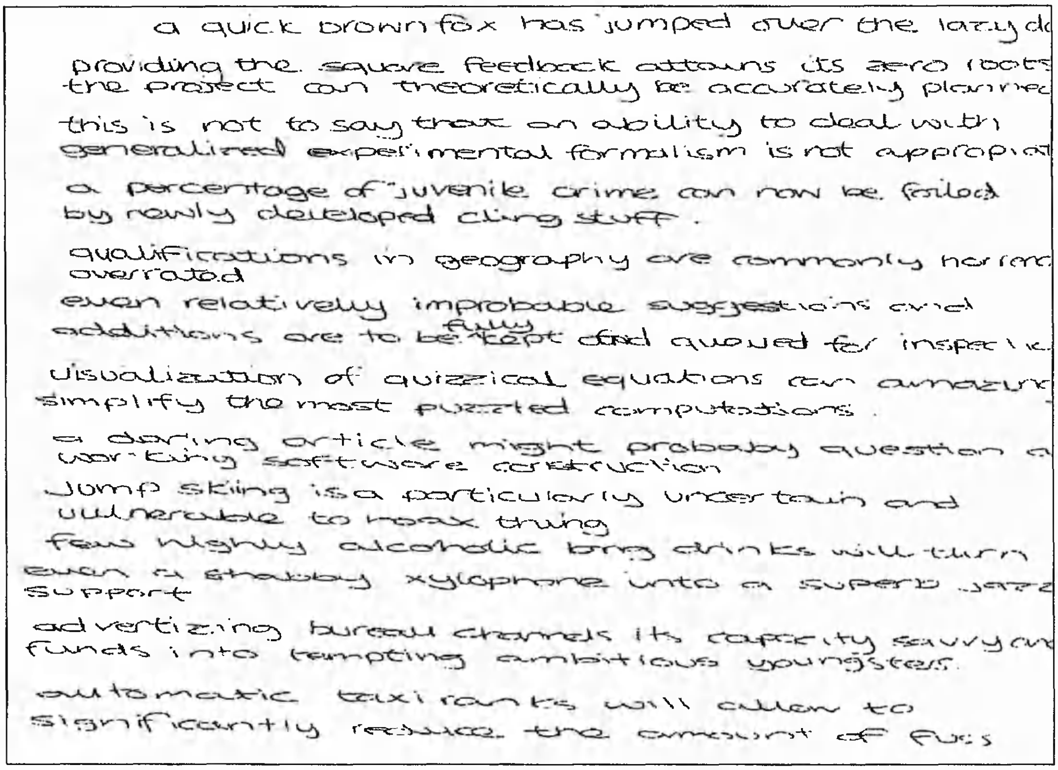


**Figure A-1**. Some samples from the database, writer 1 (lowercase)



**FigureA-2**. Some samples from the database, writer 2 (lowercase)

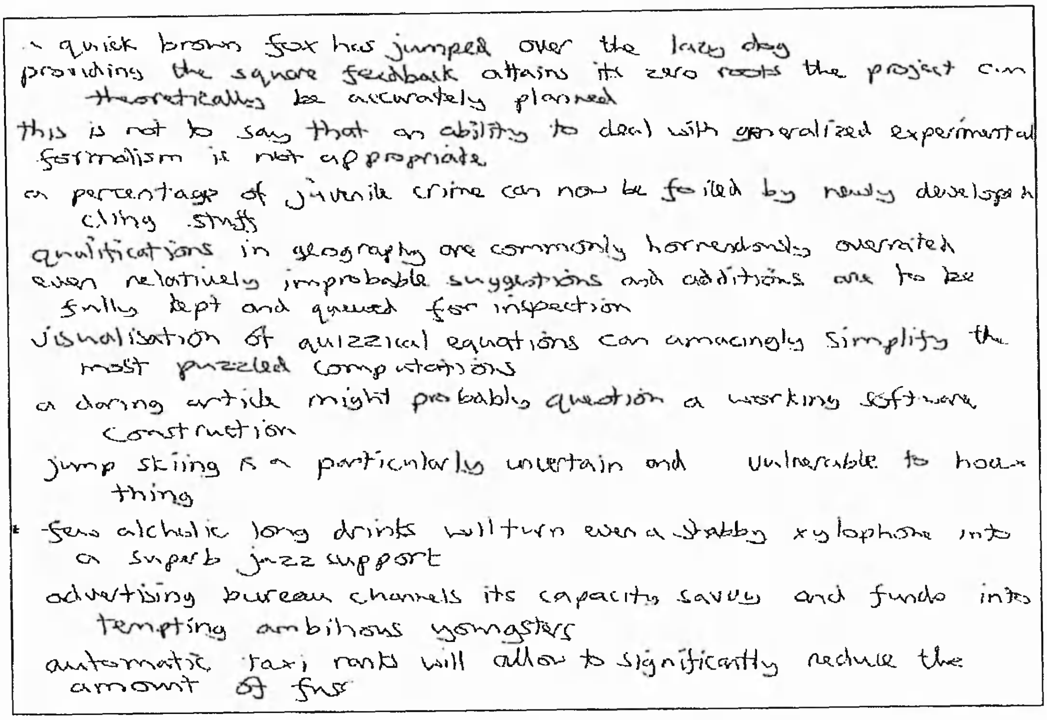


**Figure A-3**. Some samples from the database, writer 3 (lowercase)

**Figure A-4**. Some samples from the database, writer 4 (lowercase)

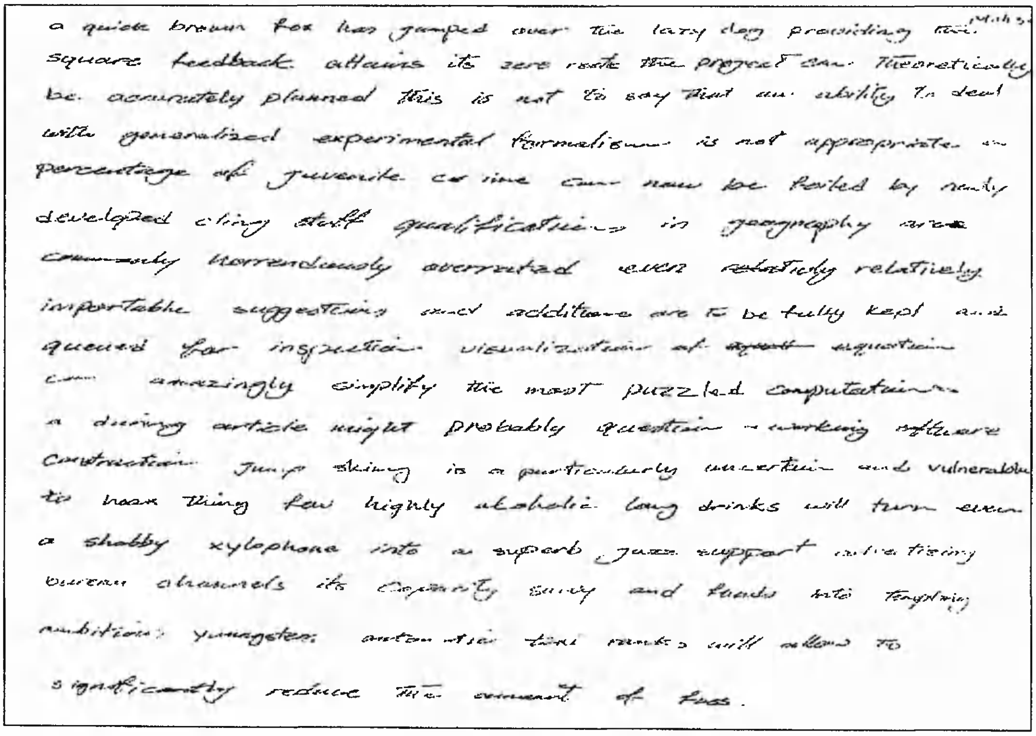



**Figure A-5**. Some samples from the database, writer 5 (lowercase)

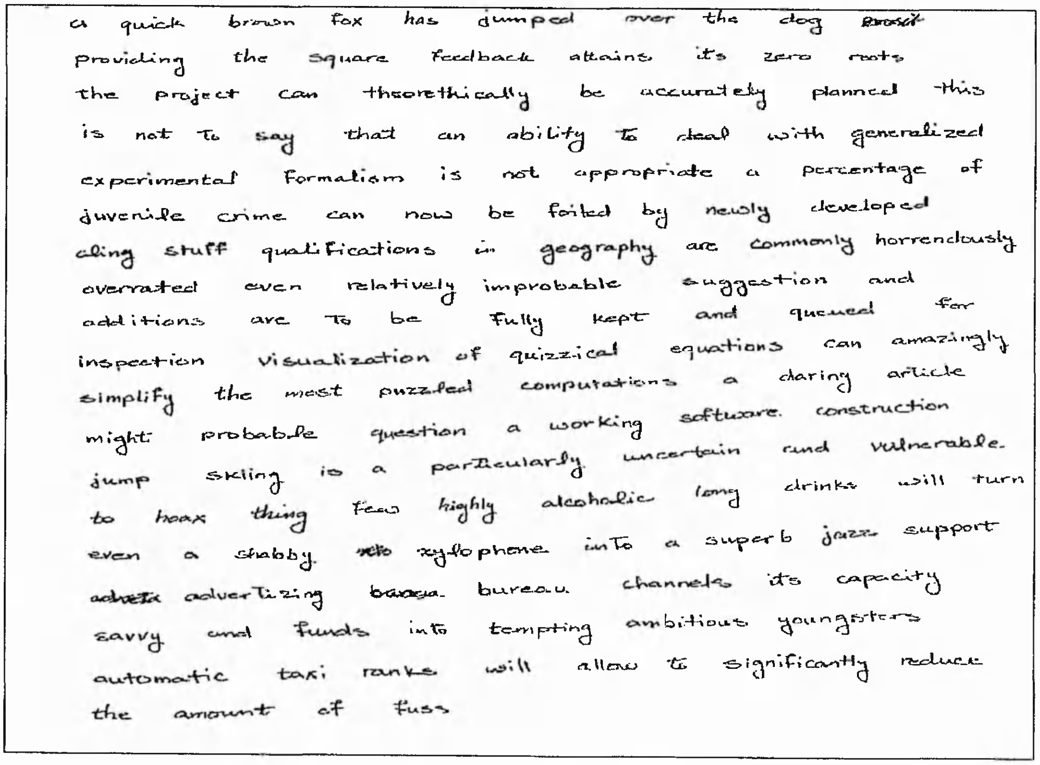



**Figure A-6**. Some samples from the database, writer 6 (lowercase)

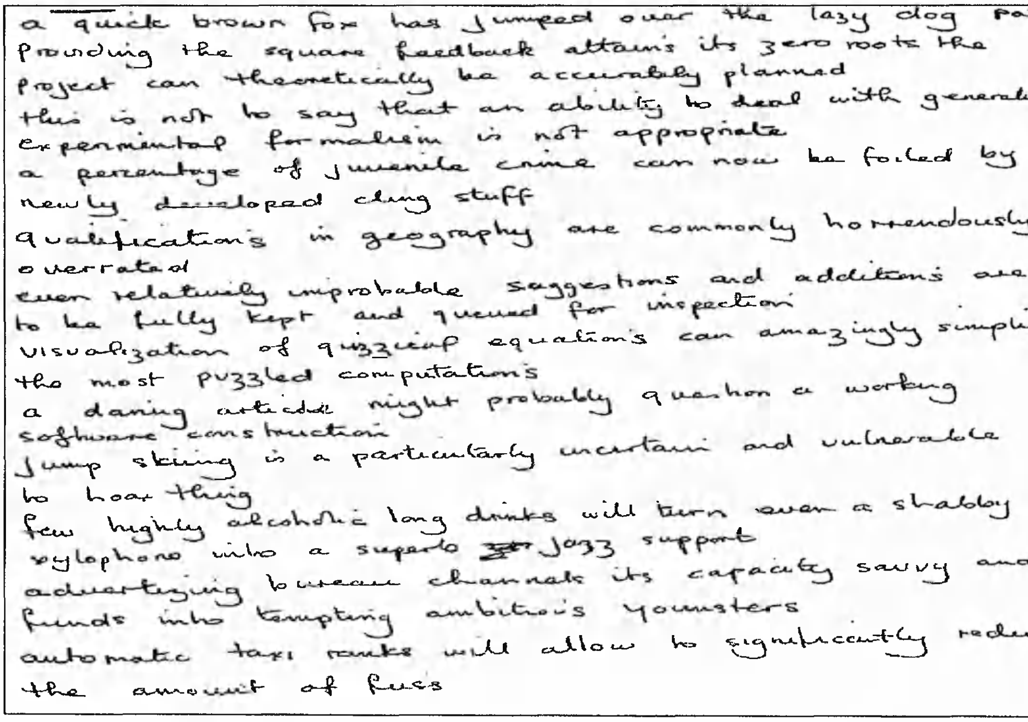



**Figure A-7**. Some samples from the database, writer 7 (lowercase)

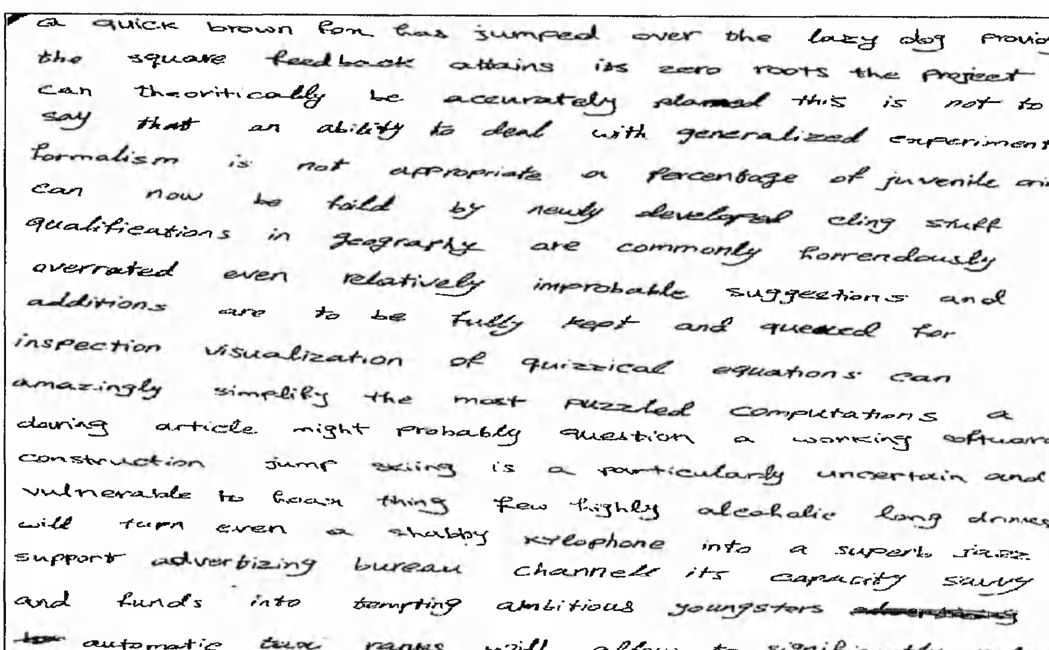



**Figure A-8**. Some samples from the database, wrier 8 (lowercase)

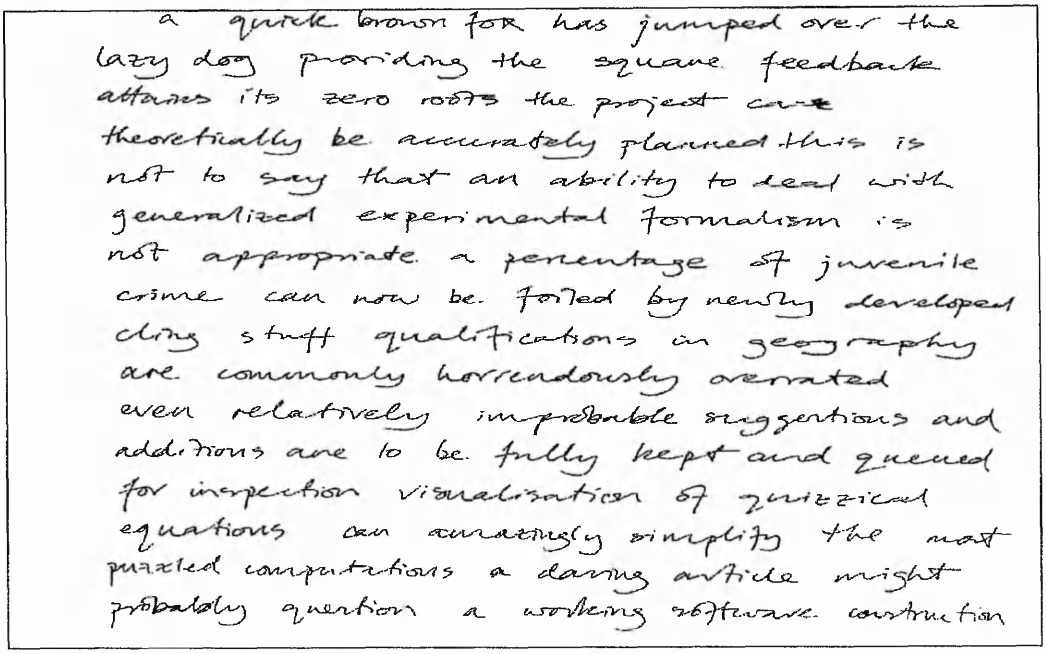



**Figure A-9**. Some samples from the database, writer 9 (lowercase)

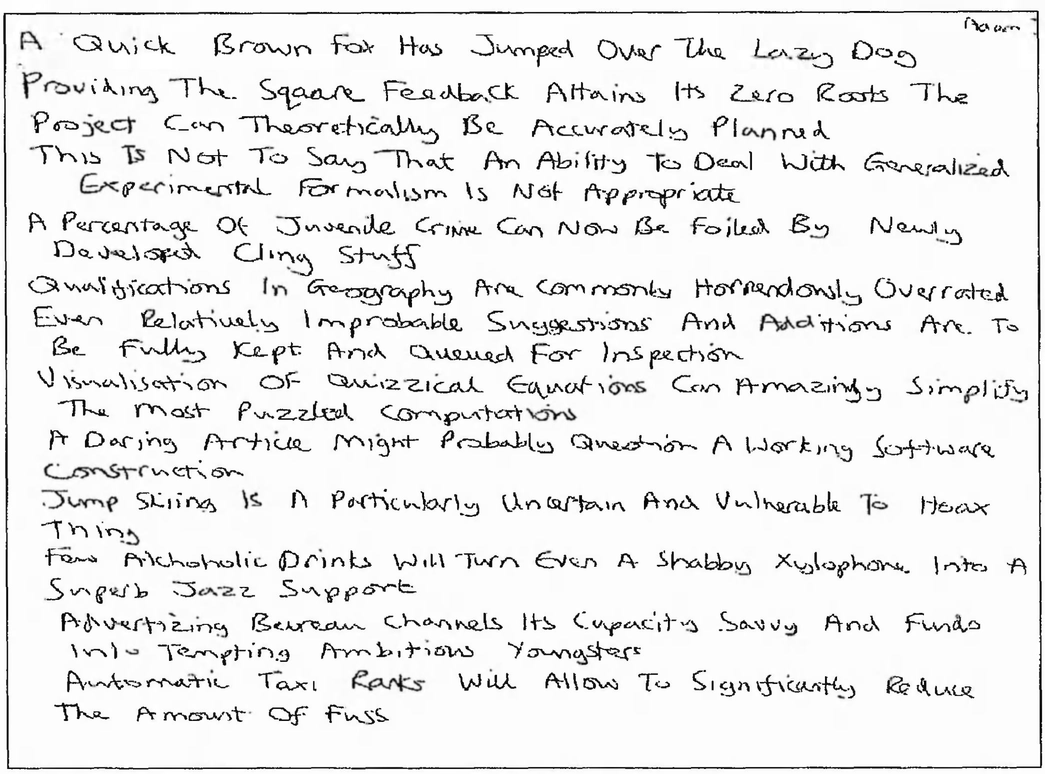



**Figure A-10**. Some samples from the database, writer 1 (mixed case)

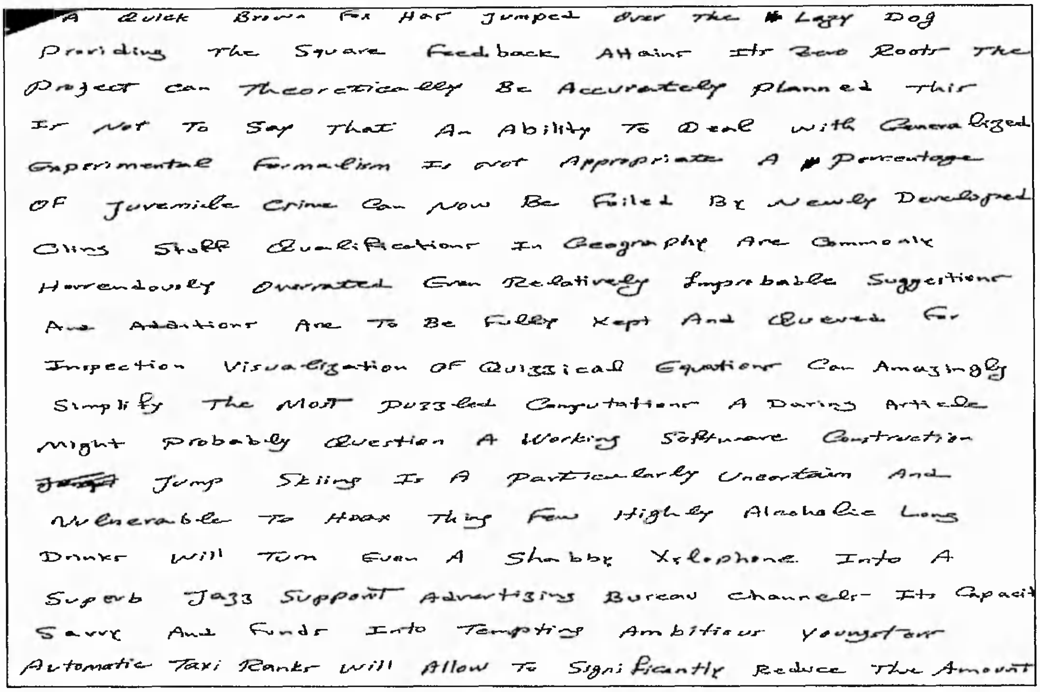


**Figure A-11**. Some samples from the database, writer 2 (mixed case)

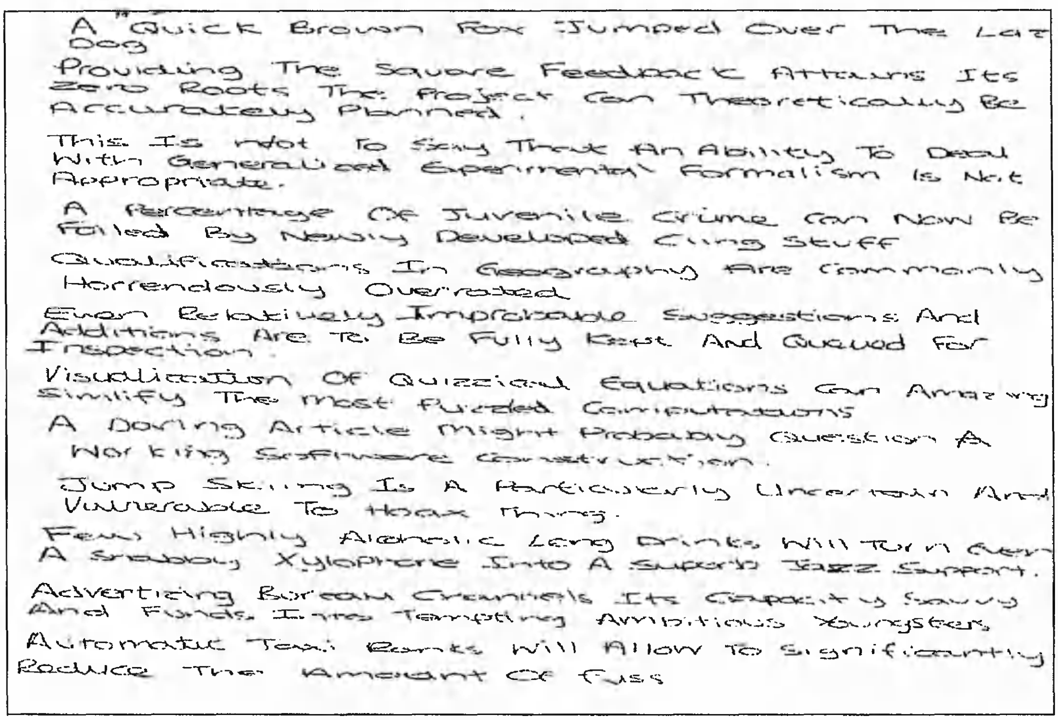


**Figure A-12**. Some samples from the database, writer 3 (mixed case)

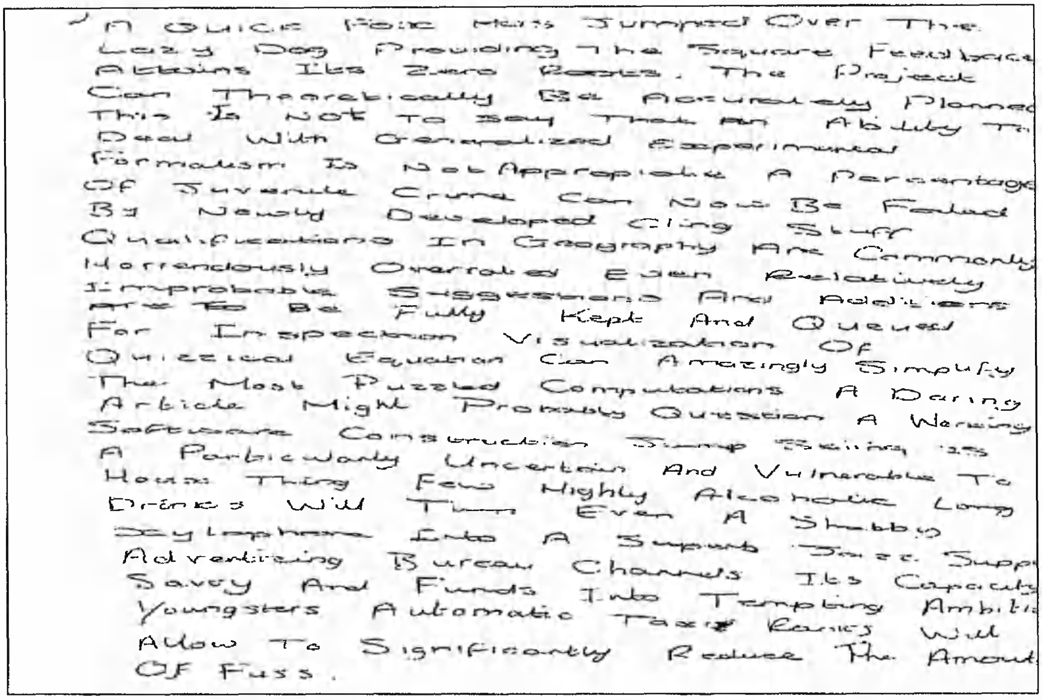



**Figure A-13**. Some samples from the database, writer 4 (mixed case)

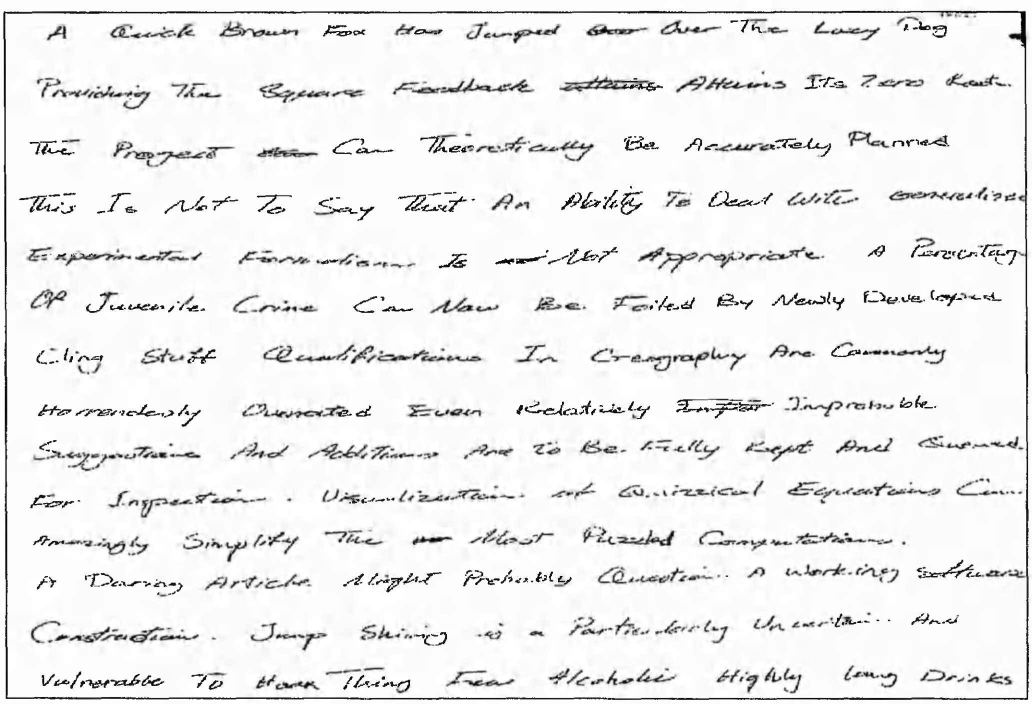



**Figure A-14**. Some samples from the database, writer 5 (mixed case)

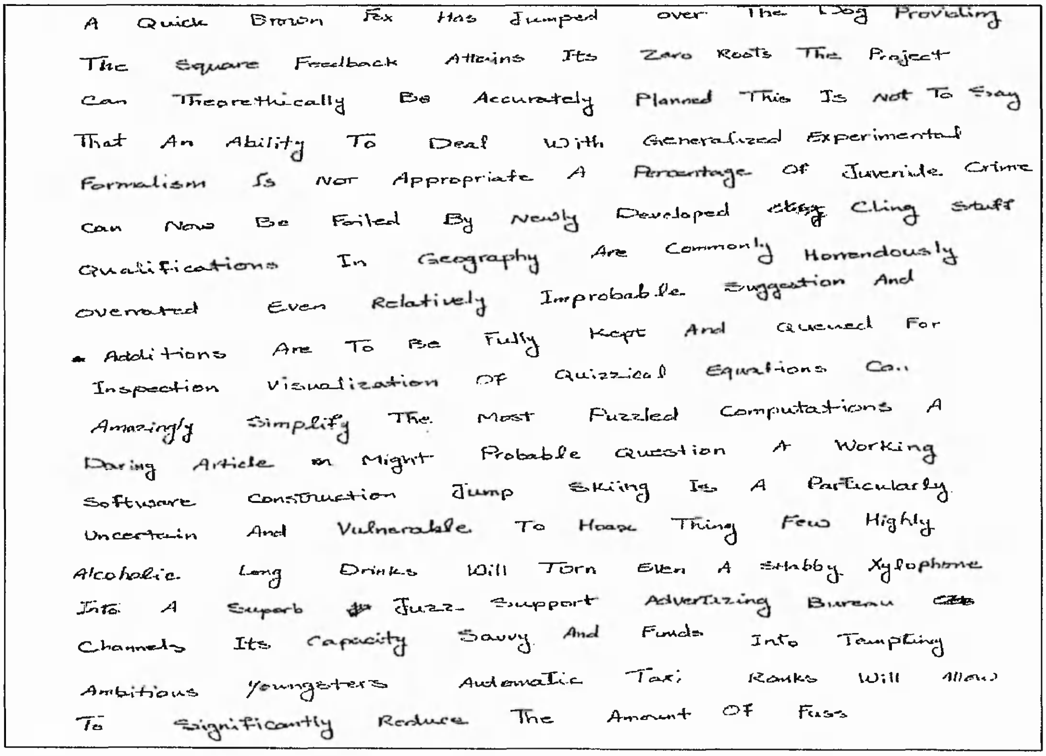



**Figure A-15**. Some samples from the database, writer 6 (mixed case)

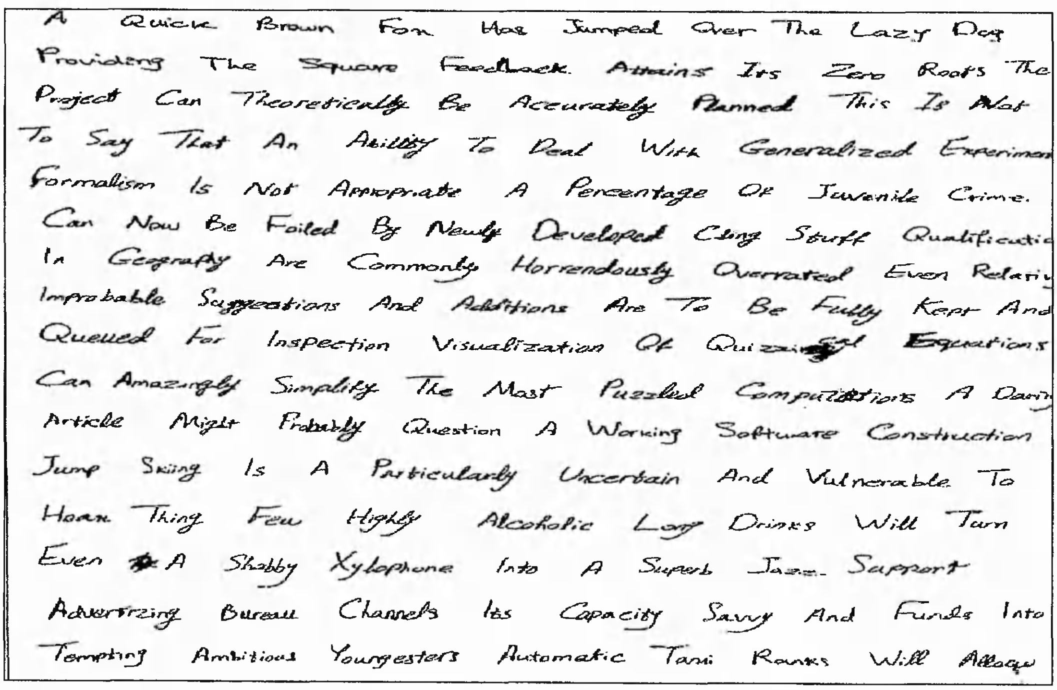



**Figure A-16**. Some samples from the database, writer 7 (mixed case)

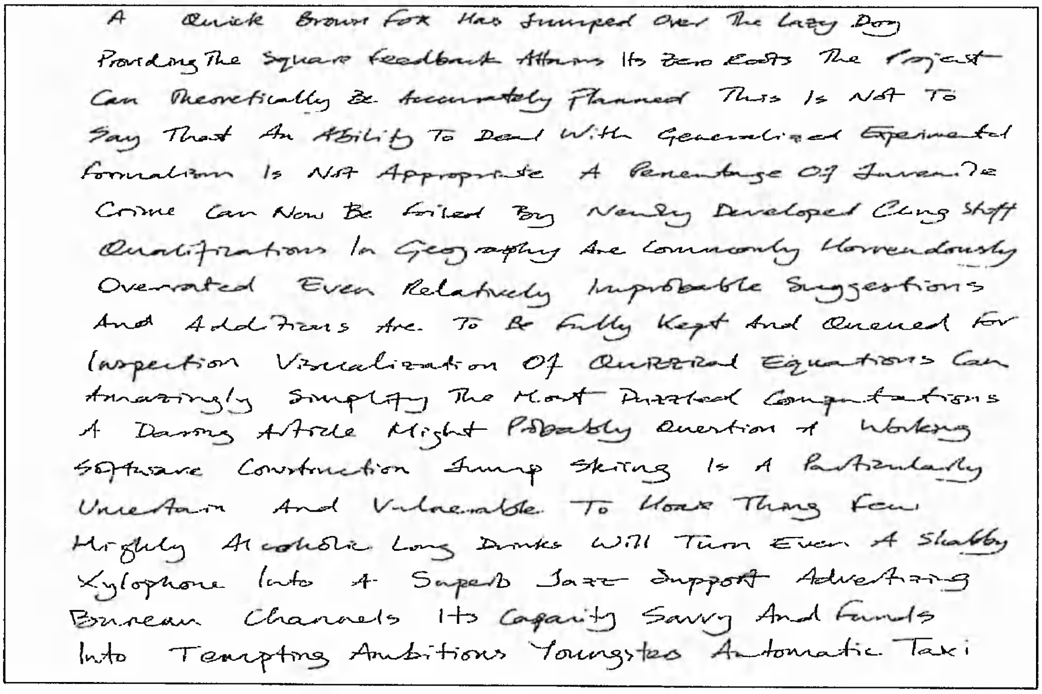



**Figure A-17**. Some samples from the database, writer 8 (mixed case)

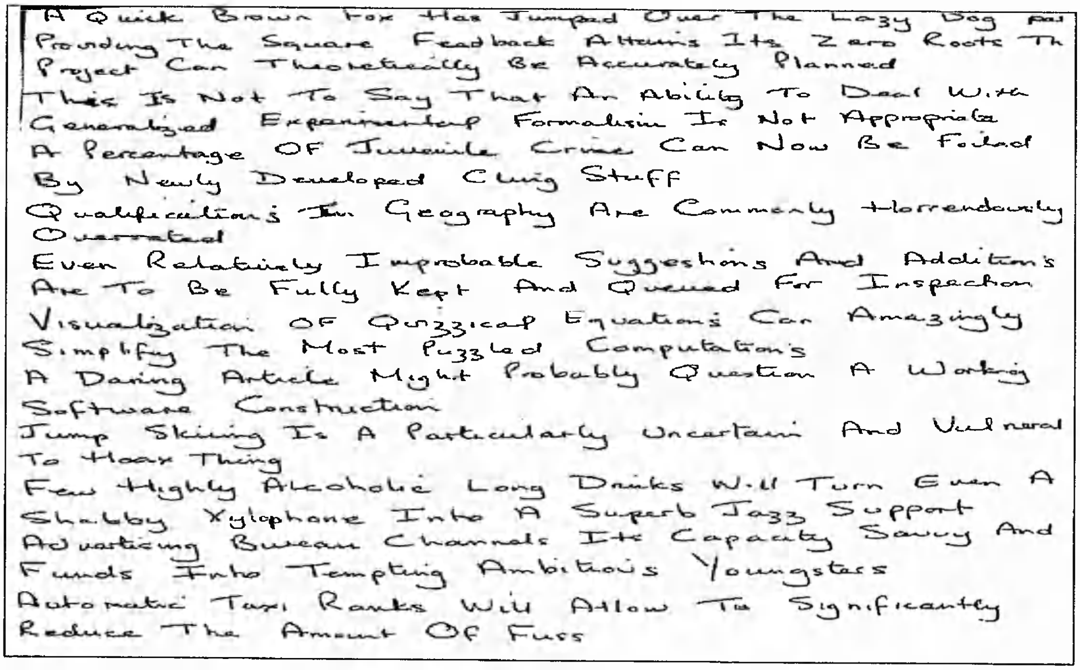



**Figure A-18**. Some samples from the database, writer 9 (mixed case)

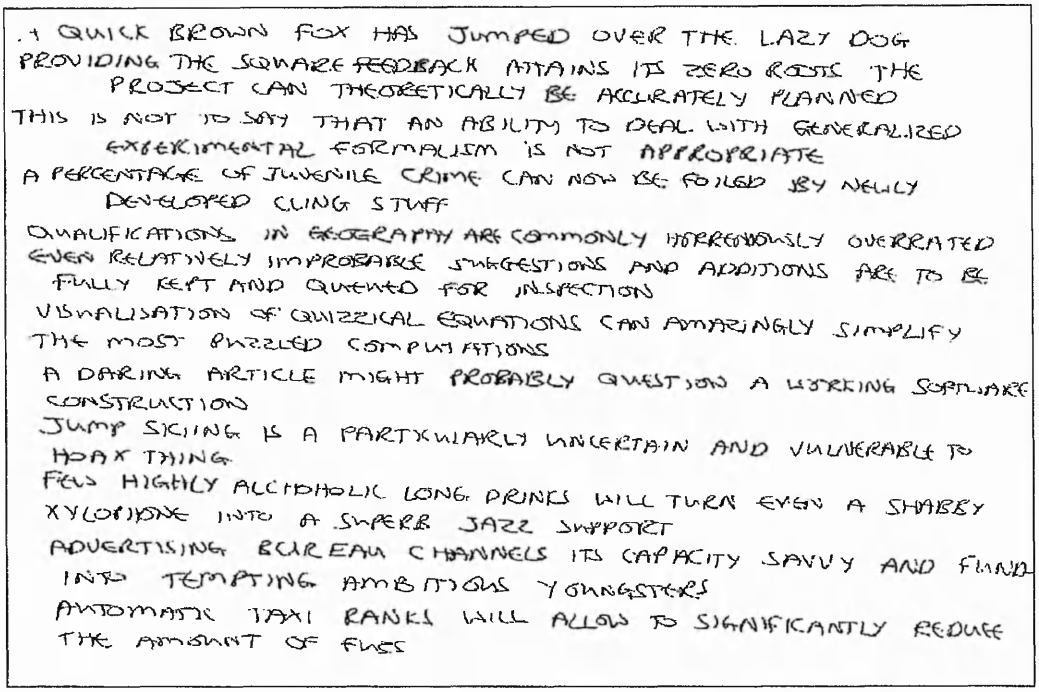



**Figure A-19**. Some samples from the database, writer 1 (upper case)

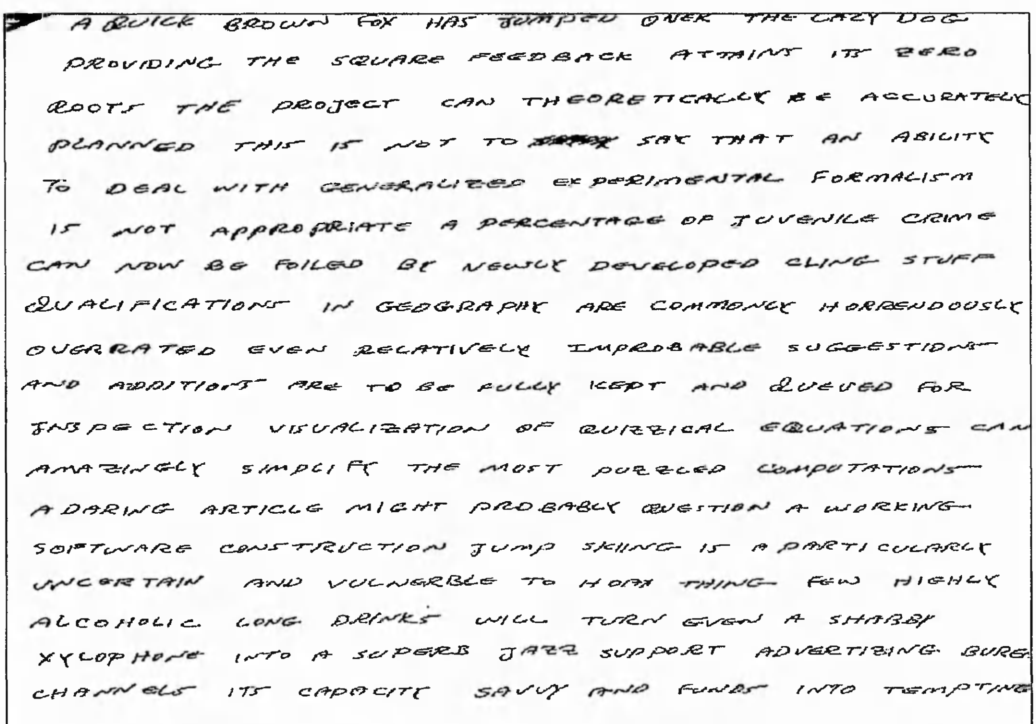



**Figure A-20**. Some samples from the database, writer 2 (upper case)

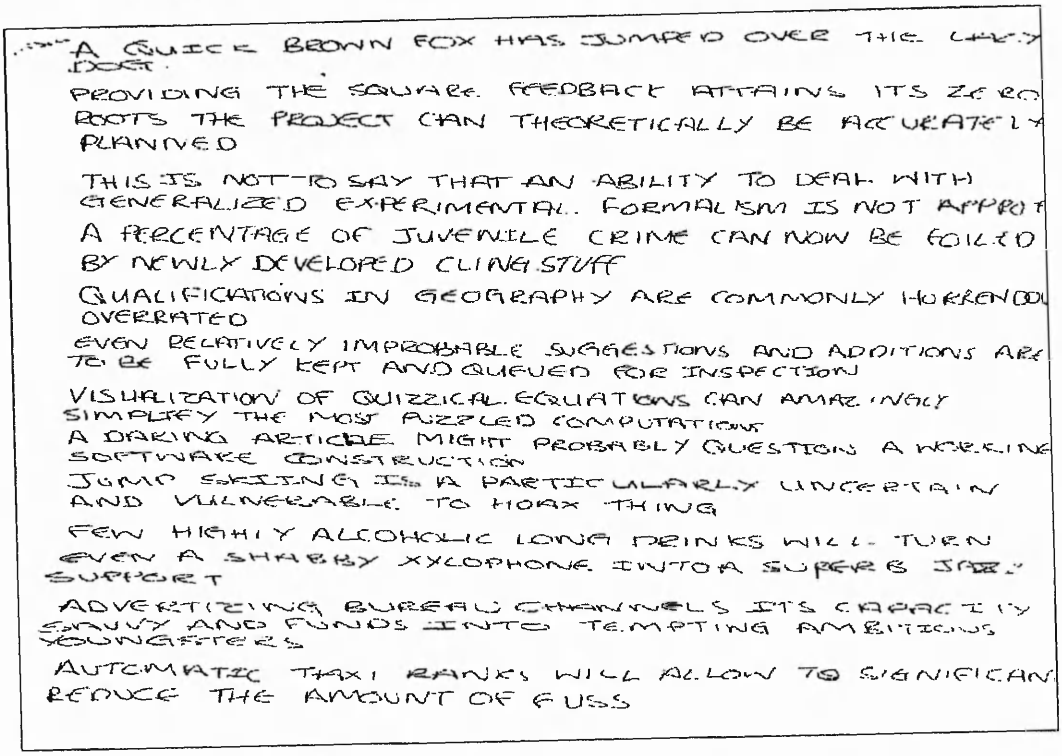



**Figure A-21**. Some samples from the database, writer 3 (upper case)

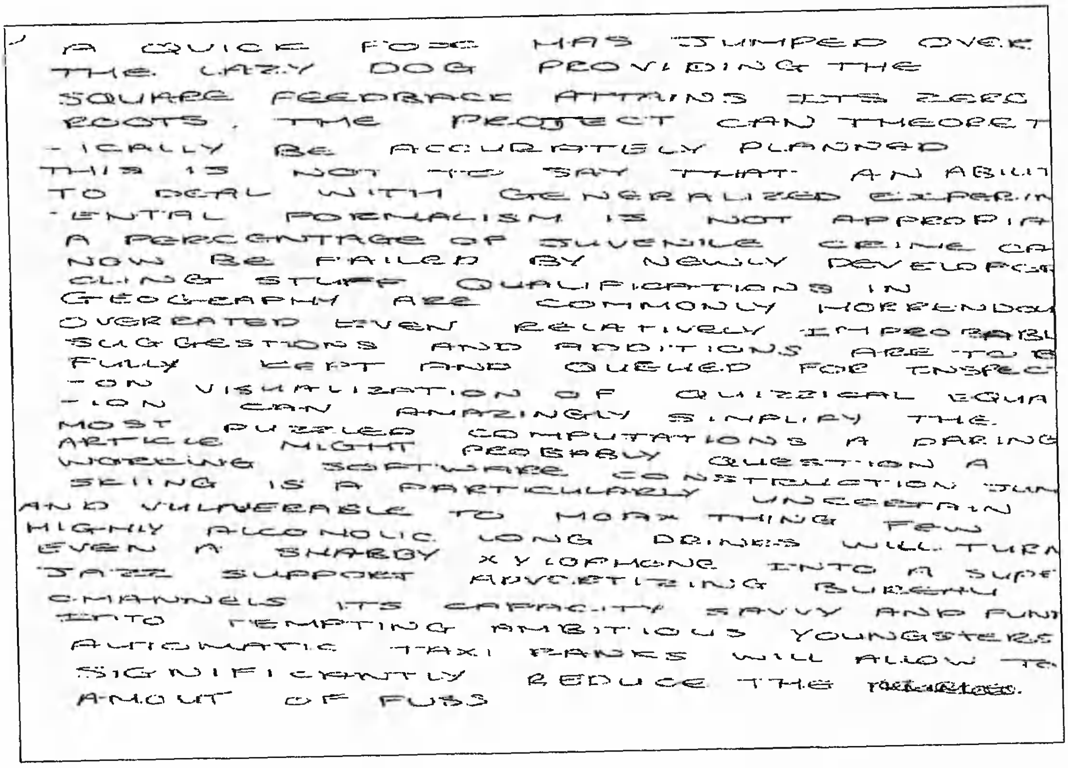



**Figure A-22**. Some samples from the database, writer 4 (upper case)

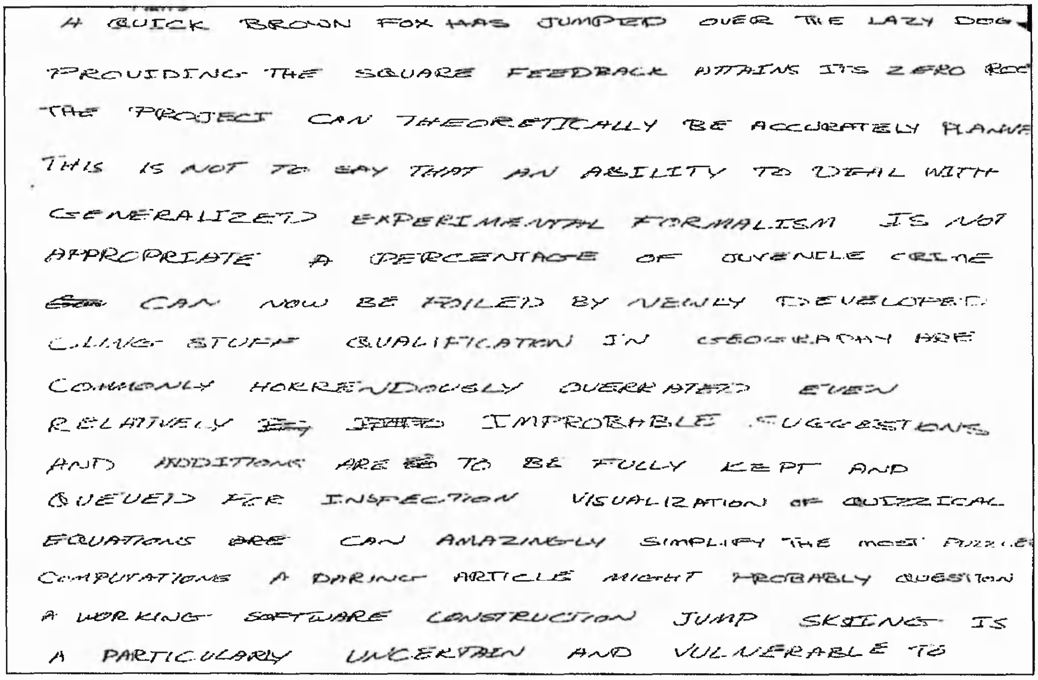



**Figure A-23**. Some samples from the database, writer 5 (upper case)

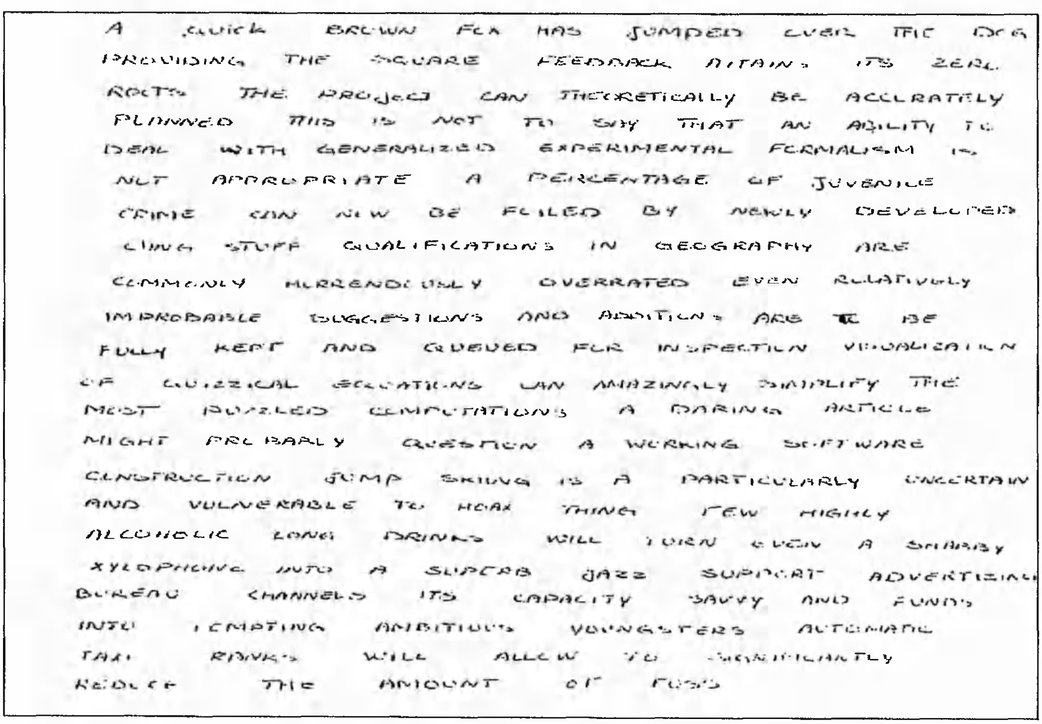
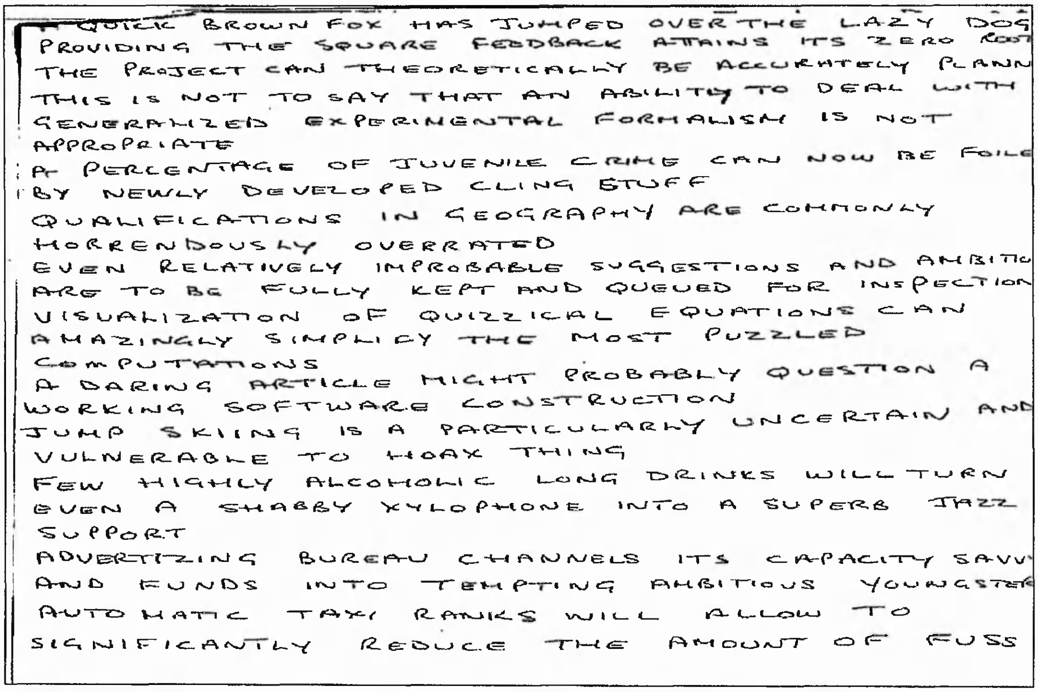
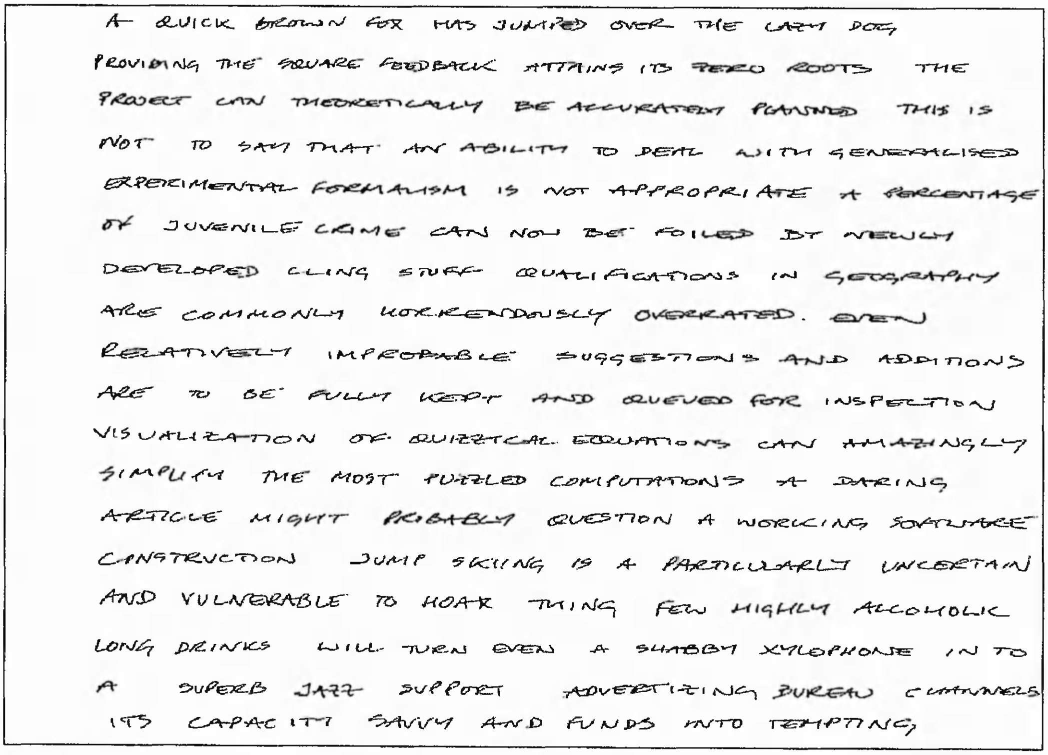



**Figure A-24**. Some samples from the database, writer 6 (upper case)



A QUICK BROWN FOX HAS JUMPED OVER THE LAZY DOG
PROVIDING THE SQUARE FEEDBACK ATTAINS ITS ZERO ROOT
THE PROJECT CAN THEORETICALLY BE ACCURATELY PLANN
THIS IS NOT TO SAY THAT AN ABILITY TO DEAL WITH
GENERALIZED EXPERIMENTAL FORMALISM IS NOT
APPROPRIATE
A PERCENTAGE OF JUVENILE CRIME CAN NOW BE FOILE
BY NEWLY DEVELOPED CLING STUFF
QUALIFICATIONS IN GEOGRAPHY ARE COMMONLY
HORRENDOUSLY OVERRATED
EVEN RELATIVELY IMPROBABLE SUGGESTIONS AND AMBITIO
ARE TO BE FULLY KEPT AND QUEUED FOR INSPECTION
VISUALIZATION OF QUIZZICAL EQUATIONS CAN
AMAZINGLY SIMPLIFY THE MOST PUZZLED
COMPUTATIONS
A DARING ARTICLE MIGHT PROBABLY QUESTION A
WORKING SOFTWARE CONSTRUCTION
JUMP SKIING IS A PARTICULARLY UNCERTAIN AND
VULNERABLE TO HOAX THING
FEW HIGHLY ALCOHOLIC LONG DRINKS WILL TURN
EVEN A SHABBY XYLOPHONE INTO A SUPERB JAZZ
SUPPORT
ADVERTIZING BUREAU CHANNELS ITS CAPACITY SAVV
AND FUNDS INTO TEMPTING AMBITIOUS YOUNGSTER
AUTOMATIC TAXI RANKS WILL ALLOW TO
SIGNIFICANTLY REDUCE THE AMOUNT OF FUSS

**Figure A-25**. Some samples from the database, writer 7 (upper case)



A QUICK BROWN FOX HAS JUMPED OVER THE LAZY DOG
PROVIDING THE SQUARE FEEDBACK ATTAINS ITS ZERO ROOTS THE
PROJECT CAN THEORETICALLY BE ACCURATELY PLANNED THIS IS
NOT TO SAY THAT AN ABILITY TO DEAL WITH GENERALISED
EXPERIMENTAL FORMALISM IS NOT APPROPRIATE A PERCENTAGE
OF JUVENILE CRIME CAN NOW BE FOILED BY NEWLY
DEVELOPED CLING STUFF QUALIFICATIONS IN GEOGRAPHY
ARE COMMONLY HORRENDOUSLY OVERRATED. EVEN
RELATIVELY IMPROBABLE SUGGESTIONS AND ADDITIONS
ARE TO BE FULLY KEPT AND QUEUED FOR INSPECTION
VISUALIZATION OF QUIZZICAL EQUATIONS CAN AMAZINGLY
SIMPLIFY THE MOST PUZZLED COMPUTATIONS A DARING
ARTICLE MIGHT PROBABLY QUESTION A WORKING SOFTWARE
CONSTRUCTION JUMP SKIING IS A PARTICULARLY UNCERTAIN
AND VULNERABLE TO HOAX THING FEW HIGHLY ALCOHOLIC
LONG DRINKS WILL TURN EVEN A SHABBY XYLOPHONE IN TO
A SUPERB JAZZ SUPPORT ADVERTIZING BUREAU CHANNELS
ITS CAPACITY SAVVY AND FUNDS INTO TEMPTING

**Figure A-26**. Some samples from the database, writer 8 (upper case)

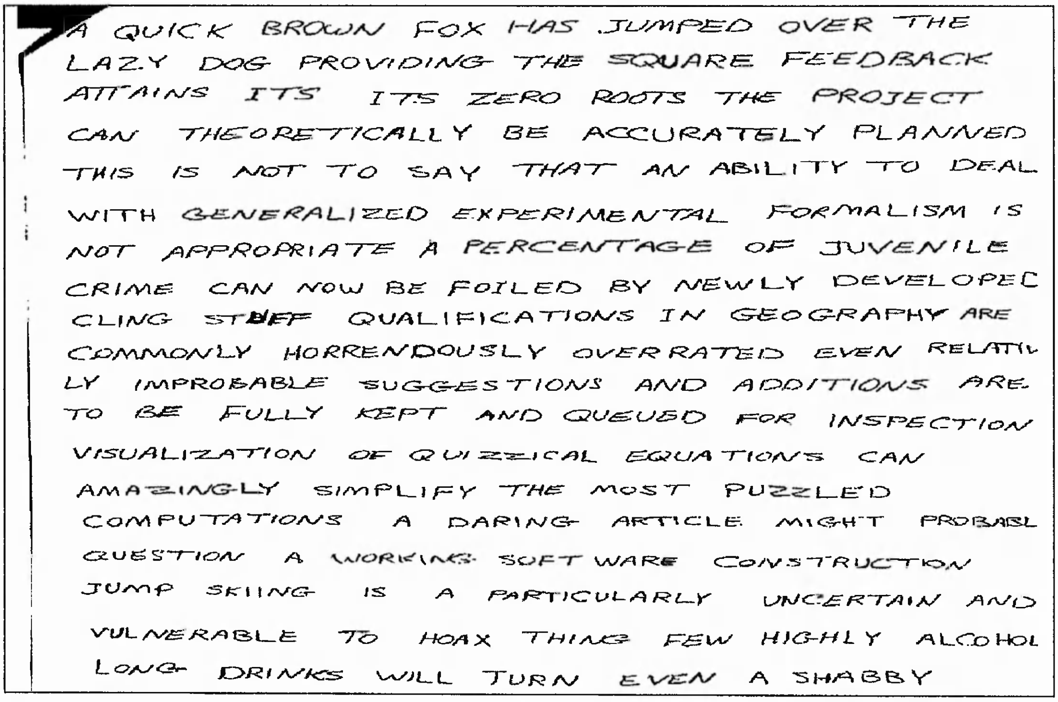

A QUICK BROWN FOX HAS JUMPED OVER THE
LAZY DOG PROVIDING THE SQUARE FEEDBACK
ATTAINS ITS ITS ZERO ROOTS THE PROJECT
CAN THEORETICALLY BE ACCURATELY PLANNED
THIS IS NOT TO SAY THAT AN ABILITY TO DEAL
WITH GENERALIZED EXPERIMENTAL FORMALISM IS
NOT APPROPRIATE A PERCENTAGE OF JUVENILE
CRIME CAN NOW BE FOILED BY NEWLY DEVELOPEC
CLING STAFF QUALIFICATIONS IN GEOGRAPHY ARE
COMMONLY HORRENDOUSLY OVERRATED EVEN RELATIV
LY IMPROBABLE SUGGESTIONS AND ADDITIONS ARE
TO BE FULLY KEPT AND QUEUED FOR INSPECTION
VISUALIZATION OF QUIZZICAL EQUATIONS CAN
AMAZINGLY SIMPLIFY THE MOST PUZZLED
COMPUTATIONS A DARING ARTICLE MIGHT PROBABL
QUESTION A WORKING SOFTWARE CONSTRUCTION
JUMP SKIING IS A PARTICULARLY UNCERTAIN AND
VULNERABLE TO HOAX THING FEW HIGHLY ALCOHOL
LONG DRINKS WILL TURN EVEN A SHABBY

**Figure A-27**. Some samples from the database, writer 9 (upper case)

# Appendix B. PNN method in case classification

## B.1 Binary classification using common $\sigma$

### B.1.1 Analysis of zoning information using 36 extracted features

Figures B-1 to B-4 show the finer details of this classification method by looking at the characteristics of the words that are misclassified. The X-axis shows the word zoning information. It is assumed that lower case words can reside in 1,2 and 3 zones, mixed words can occupy only 2 and 3 zones and upper words can reside one zone only.

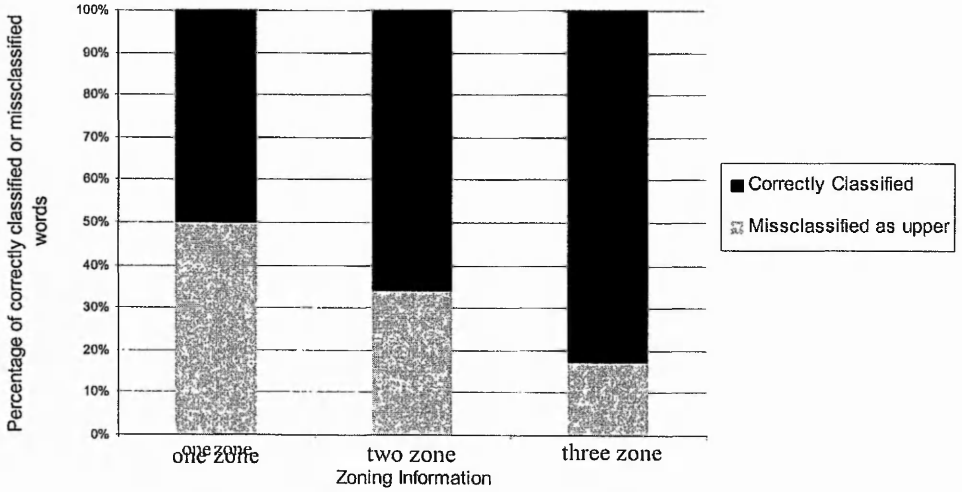



**Figure B-1:** Percentage of lower case words, which are correctly or incorrectly classified (lower/upper data set) using all 36 features.

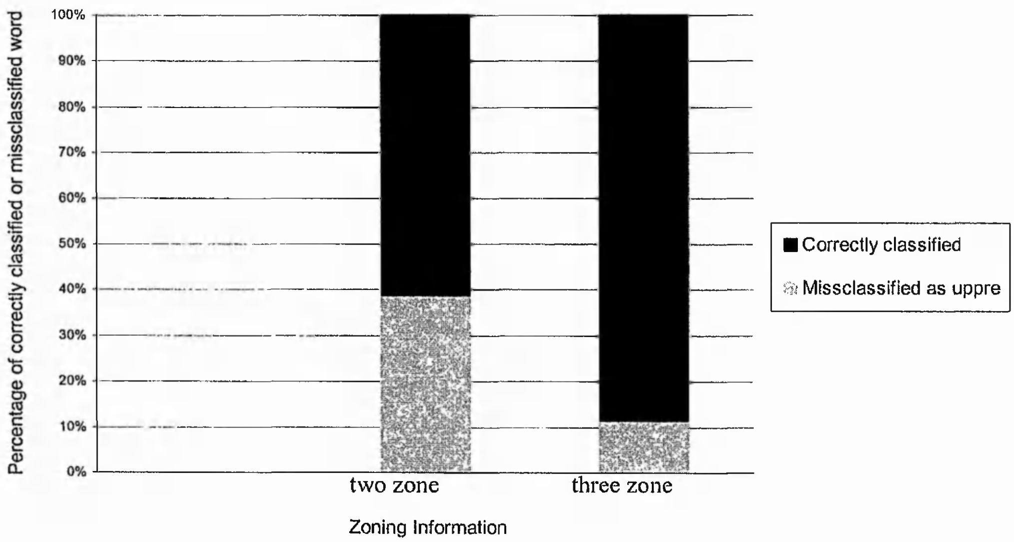



**Figure B-2:** Percentage of mixed words, which are correctly or incorrectly classified (upper/mixed data set) using all 36 features.

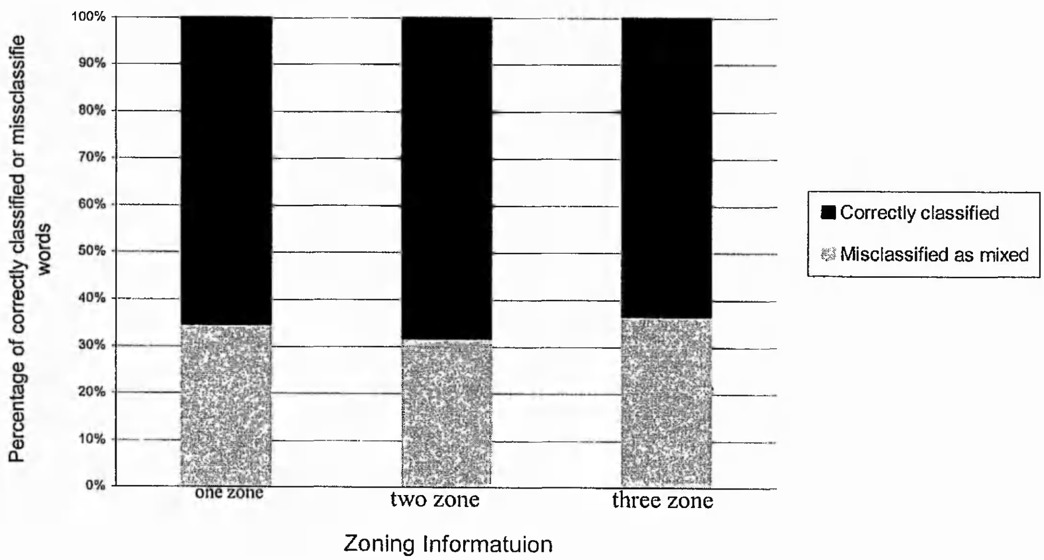



**Figure B-3:** Percentage of lower words, which are correctly or incorrectly classified (lower/mixed data set) using all 36 features.

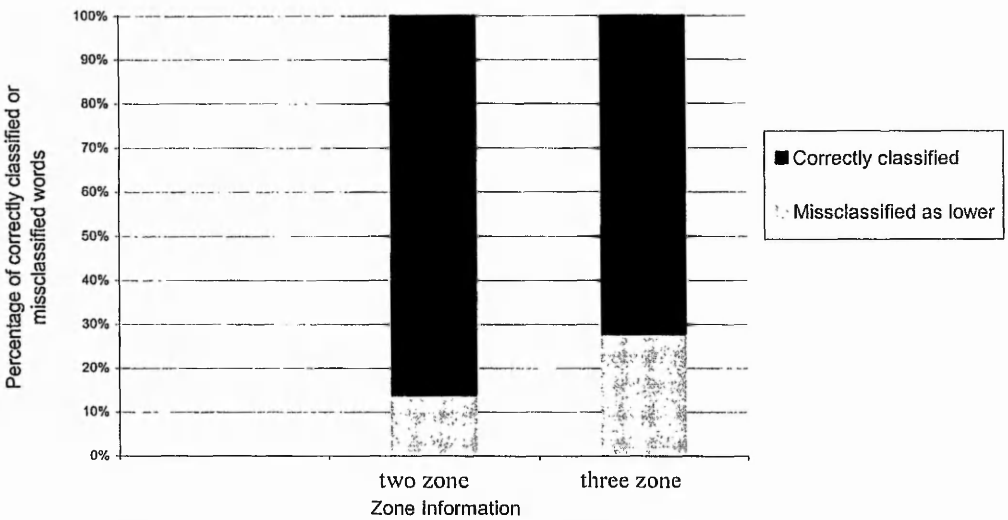



**Figure B-4:** Percentage of mixed words, which are correctly or incorrectly classified (lower/mixed data set) using all 36 features.

Figure B-1 shows that in a classification between lower and upper case images 50.00%, 65.91% and 84.62% of one, two and three zones lower words images are correctly classified as lowercase words with the rest of the words being misclassified as upper case words. Not surprisingly, this shows that the number of one zone lower case words "even", "are", etc which are misclassified as upper case words is greater than the number of two and three zone word images that are misclassified. The number of zones occupied by a word is thus a crucial factor in differentiation between lower case and upper case words.

This effect is mirrored in the results shown in figure B-2 where 61.22% and 88.46% of two and three zone mixed case words are correctly classified as mixed case with the rest being misclassified as upper case.

Conversely, figure B-3 shows that 65.38%, 68.18% and 63.44% of one, two and three zone lower case words are correctly classified as lower case in classification between lower and mixed case images. Figure B-4 also shows that 86.15% and 72.22% of two and three zone mixed words are correctly classified in classification between lower and mixed case word.

### B.1.2 Analysis of zoning information using the selected features

Figure B-5 and B-6 again shows the finer detail of this experiment by looking at the characteristics of the words that are misclassified.

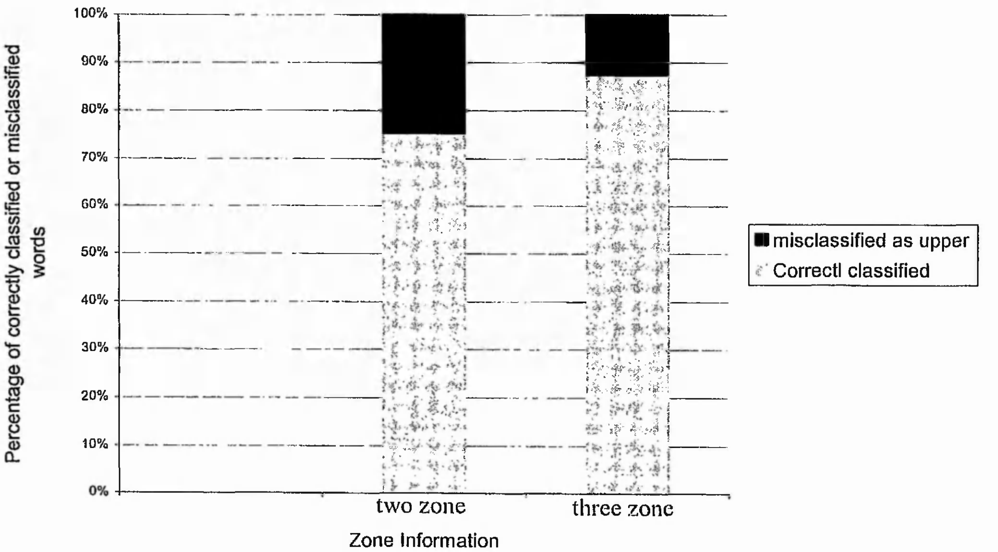



**Figure B-5:** Percentage of mixed words, which are correctly or incorrectly classified (upper/mixed data set) using 15 selected features

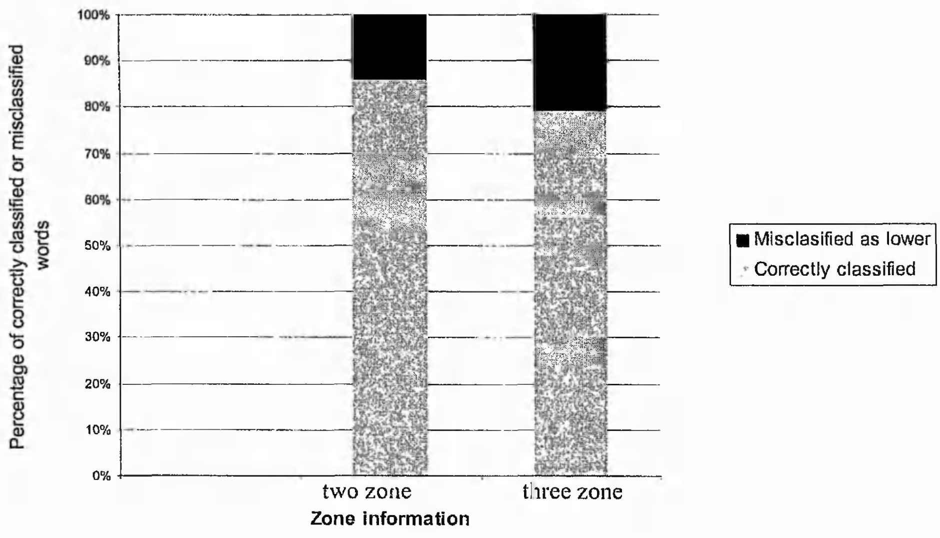



**Figure B-6:** Percentage of mixed words, which are correctly or incorrectly classified (lower/mixed data set) using 23 selected features.

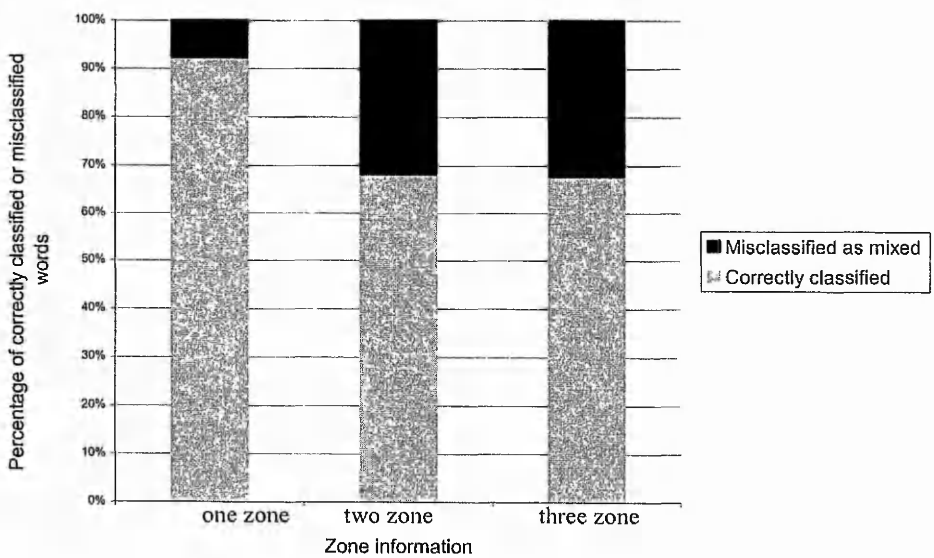



**Figure B-7:** Percentage of lower words, which are correctly or incorrectly classified (lower/mixed data set) using 23 selected features.

Figure B-5 shows that 25.00% and 12.50% of two and three zone mixed word images are misclassified as upper case words with the rest being correctly classified.

Figure B-6 shows that 14.00% and 21.00% of two and three zone mixed word images are misclassified as lower case words with the rest being correctly classified.

Finally Figure B-7 shows that 8.00%, 32.00% of and 32.00% one, two and three zone word images are misclassified as mixed case word with the rest being correctly classified.

## B.2 Triple classification using common $\sigma$

### B.2.1 Analysis of zoning information using 36 extracted features

These results can be broken down into finer detail by looking at the characteristics of the words that are misclassified.

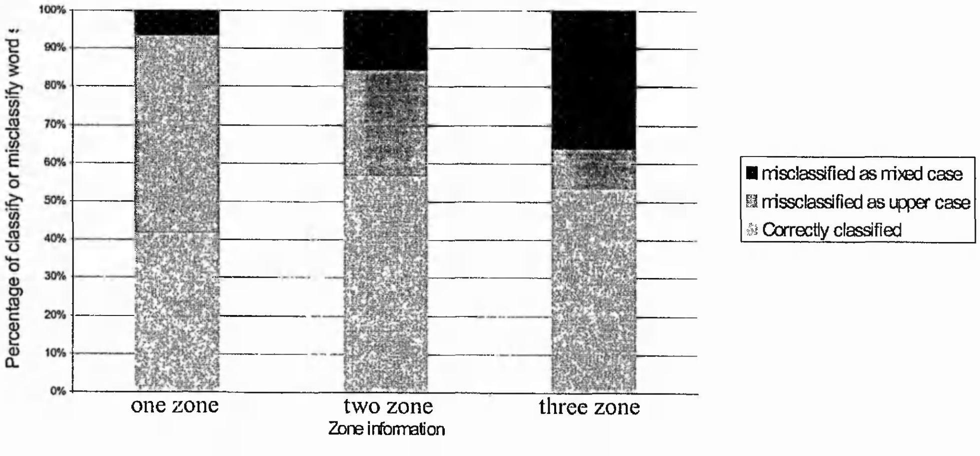



**Figure B-8:** Percentage of lower case words, which are correctly or incorrectly classified (upper/lower/mixed data set) using 36 features.

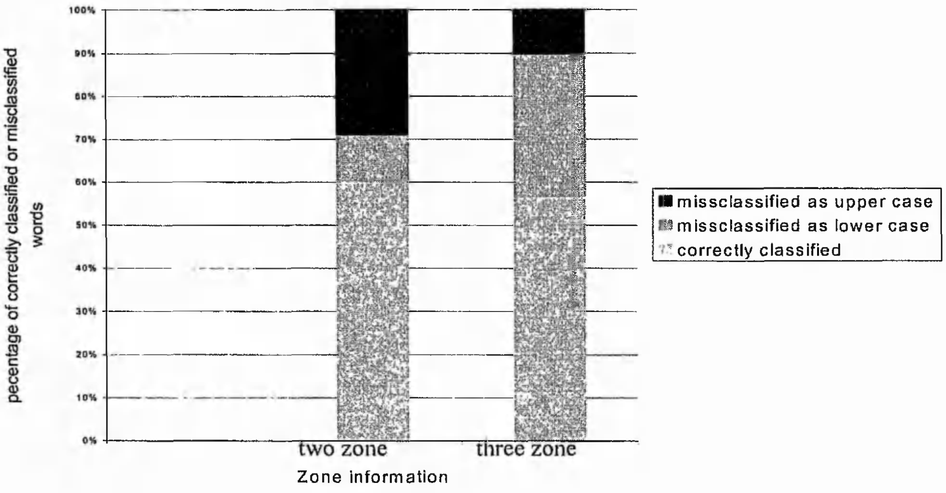



**Figure B-9:** Percentage of mixed case words, which are correctly or incorrectly classified (upper/lower/mixed data set) using 36 features.

Figure B-8 shows that 52.00%, 28.00% and 10.00% of one, two and three zone lower case word images are misclassified as upper case word images. Whilst 7.00%, 16.00%, 36.00% of one, two and three zone lower case word images are misclassified as mixed case word images respectively. This shows that the majority of the lower case words, which are misclassified as an upper case word are one zone only words such as "crime", "even", etc. This is similar to the result for binary classification and again shown the important of zoning for correct case classification. Figure B-8 also shows that most of lower case word images that are misclassified as mixed case words are three zones words such as "probably", "experimental", etc.

Figure B-9 shows that 11.00% and 33.00% of two and three zone mixed case word images are misclassified as lower case word images. Whilst 29.00% and 10.00% of two and three zone mixed case word images are misclassified as upper case word images. This shows that the majority of the mixed case words which are misclassified as upper case words are two zones words such as "Planned", "Channel", etc. Most of the mixed case word images that are misclassified as lower case words are 3 zone words such as "Probably", "Shabby", etc.

### B.2.2 Analysis of zoning information using the selected features

These results can be broken down into finer detail by looking at characteristics of the words that are misclassified. It assumes that lowercase words are located in 1, 2 and 3 zone, mixed case words are located in 2 and 3 zones and upper case words are located in one zone only.

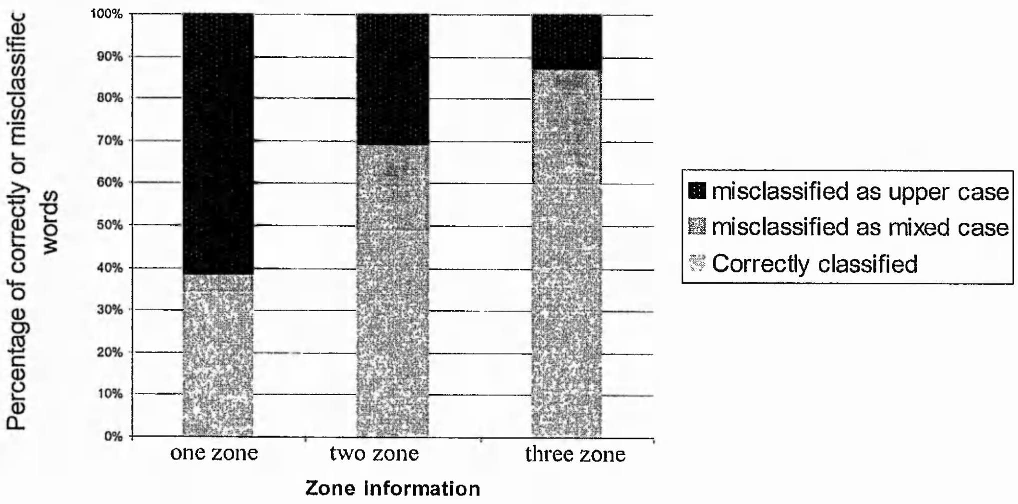



**Figure B-10:** Percentage of lower case words, which are correctly or incorrectly classified (upper/lower/mixed data set) using 25 selected features.

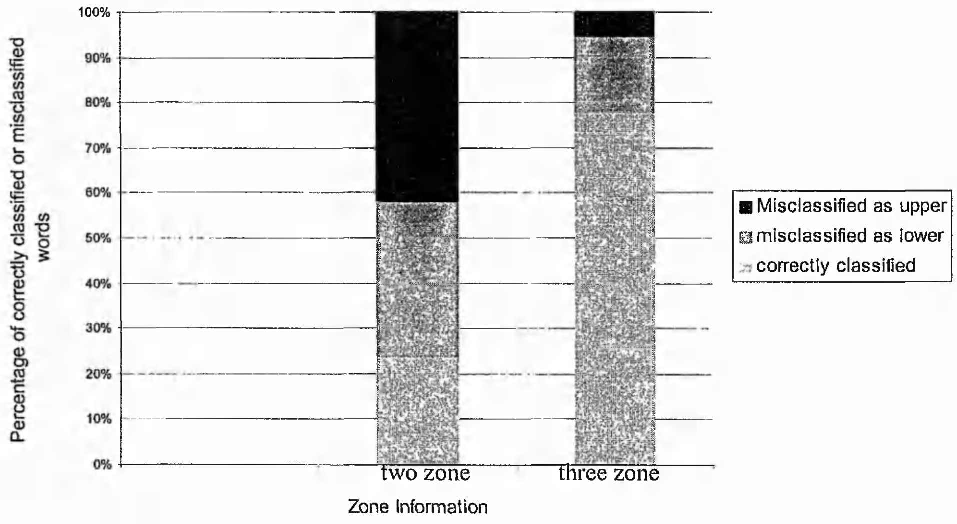



**Figure B-11:** Percentage of mixed case words, which are correctly or incorrectly classified (upper/lower/mixed data set) using 25 selected features.

Figure B-10 shows that 3.00%, 20.00% and 27.00% of one, two and three zone lower case word images are misclassified as mixed case whilst 62.00%, 31.00% and 13.00% of one, two and three lower case zone words misclassified as upper case word images.

Figure B-11 shows that 35.00% and 22.00% of two and three zone mixed case word images are misclassified as lower case words whilst 42.00% and 7.00% of two and three zone mixed case word images are misclassified as upper case words.

## B.3 Binary classification using different $\sigma_i$

### B.3.1 Analysis of zoning information using 36 extracted features

These results can be broken down into finer detail by looking at characteristics of those words that are misclassified.

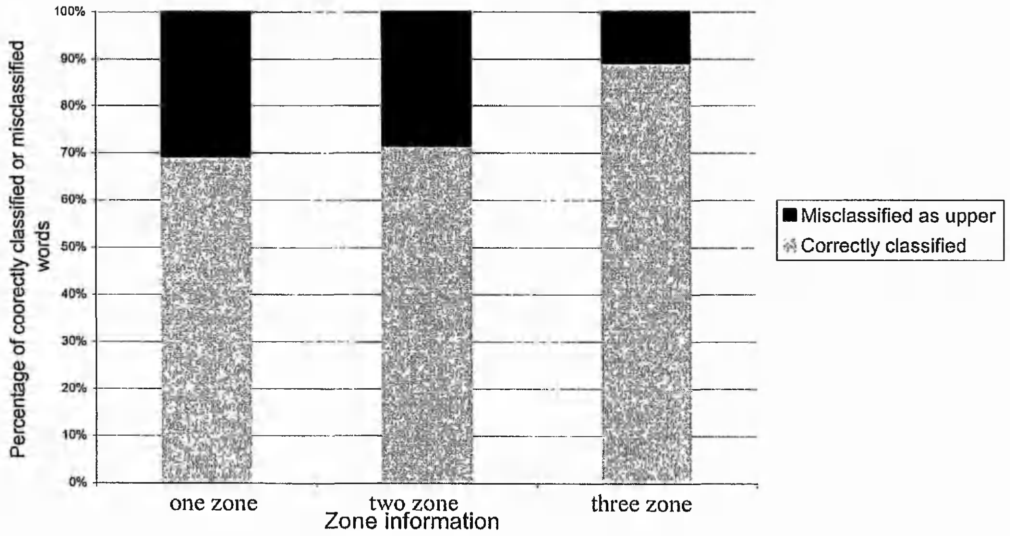



**Figure B-12:** Percentage of lower case words, which are correctly or incorrectly classified (lower/upper data set) using 36 features.

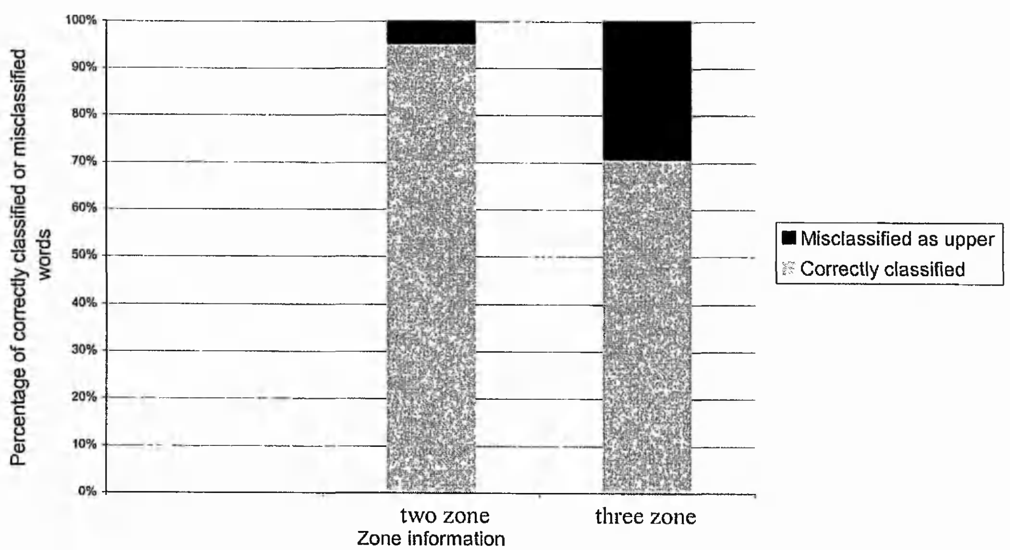



**Figure B-13:** Percentage of mixed case words, which are correctly or incorrectly classified (mixed/upper data set) using 36 features.

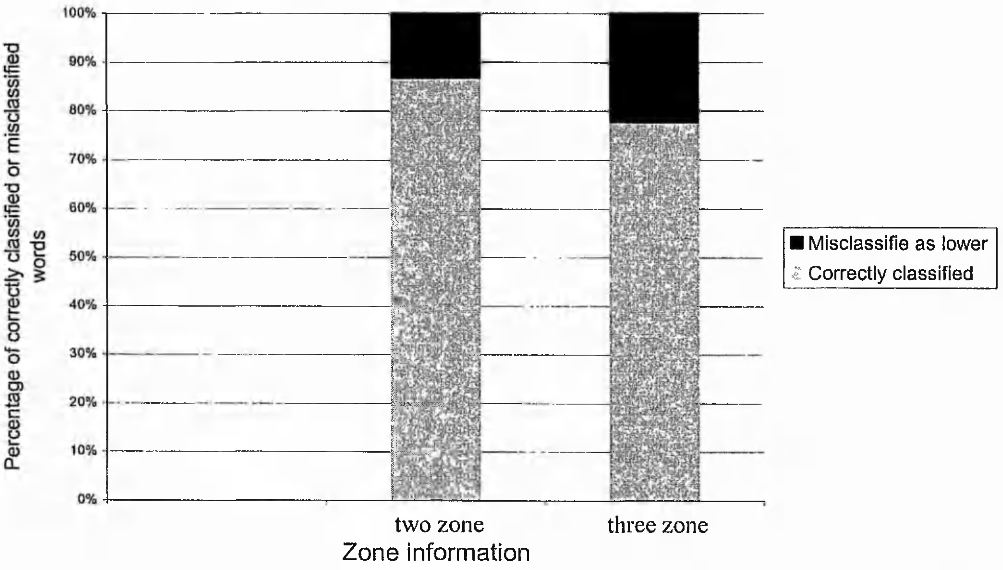



**Figure B-14:** Percentage of mixed words, which are correctly or incorrectly classified or misclassified (lower/mixed data set) using 36 features.

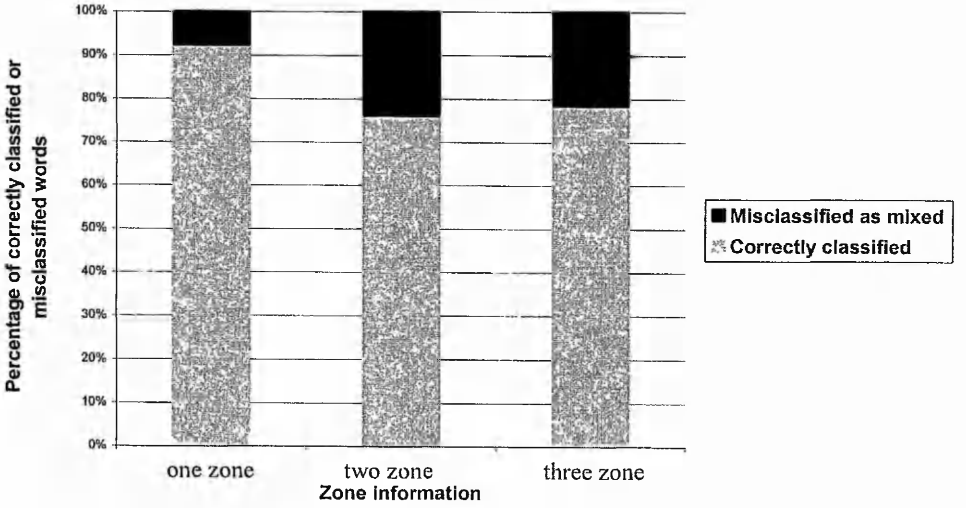



**Figure B-15:** Percentage of lower words, which are correctly or incorrectly classified (lower/mixed data set) using 36 features.

Figures B-12 shows that 31.00%, 25.00%, 11.00% of one, two and three zone lower case word images are misclassified as upper case words with the rest being correctly classified. Figure B-13 shows that 4.60% and 29.00% of two and three zone mixed case words images are misclassified as upper case words with the rest being correctly classified. Figure B-14 shows that 13.00% and 22.00% of two and three zone mixed case words are misclassified as lower case words with the rest being correctly classified. Figure B-15 shows that 8.00%, 24.00% and 21.00% of one, two, three zone lower case word images are misclassified as mixed case words and the rest are correctly classified.

### B.3.2 Analysis of zoning information using the selected features

This can broken down into detail by looking at the characteristic of those words that are misclassified.

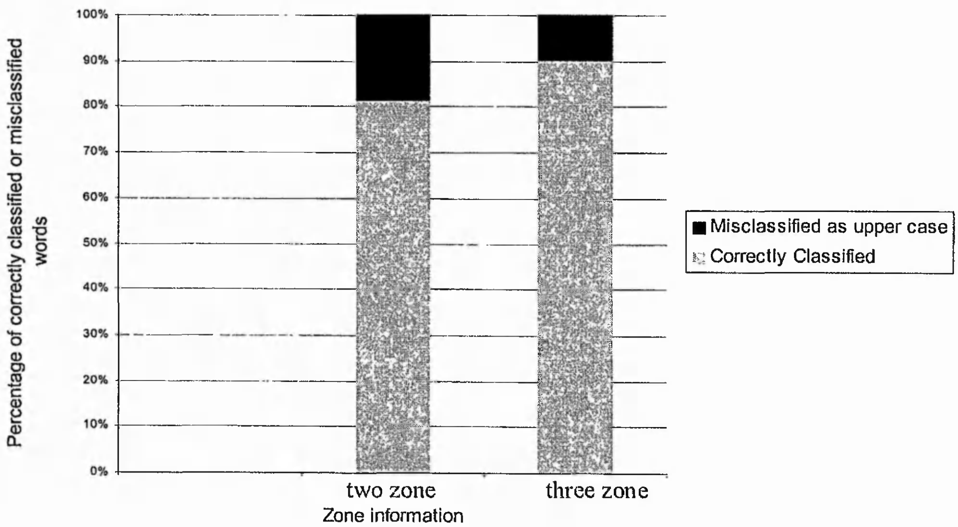



**Figure B-16:** Percentage of mixed case words, which are correctly or incorrectly classified (mixed/upper data set) using selected features.

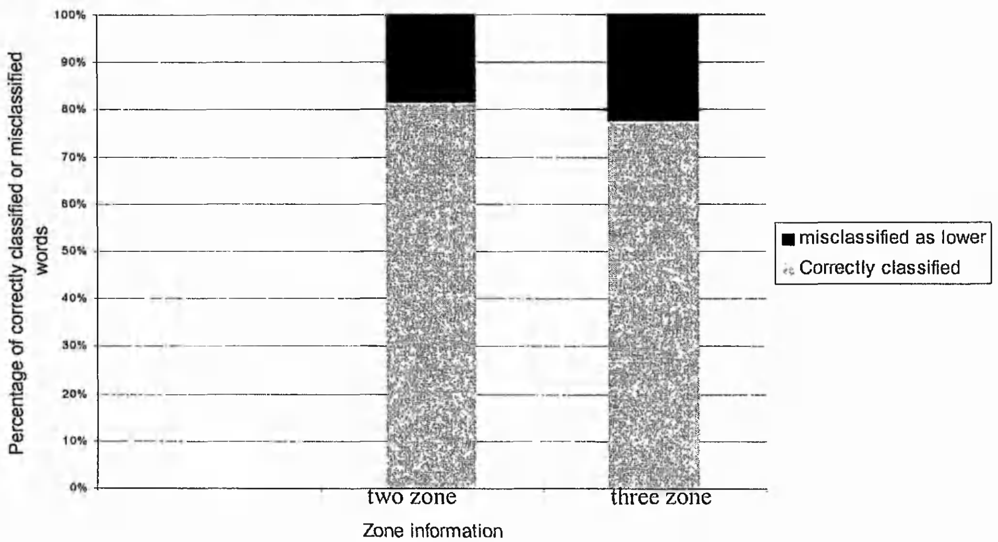



**Figure B-17:** Percentage of mixed case, which are correctly or incorrectly classified (lower/mixed data set) using selected features

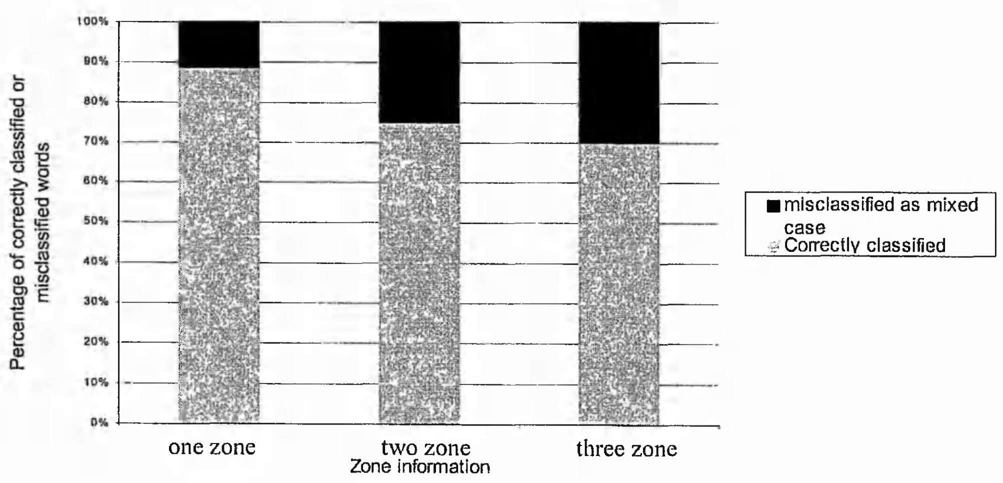



**Figure B-18:** Percentage of lower case words, which are correctly or incorrectly classified (lower/mixed data set) using selected features.

Figure B-16 shows 18.50% and 9.70% of two and three zone mixed case words are misclassified as upper case words. Figure B-17 shows 18.46% and 22.22% of one and two zone mixed case words are misclassified as lower case words. Figure 5-18 shows that 11.50%, 25.00% and 30.00% of one, two and three zone word images are misclassified as mixed case words.

## B.4 Triple Classification using different $\sigma_i$

### B.4.1 Analysis of zoning information using 36 features

These results can be is broken down to looking at the characteristics of the words that are misclassified.

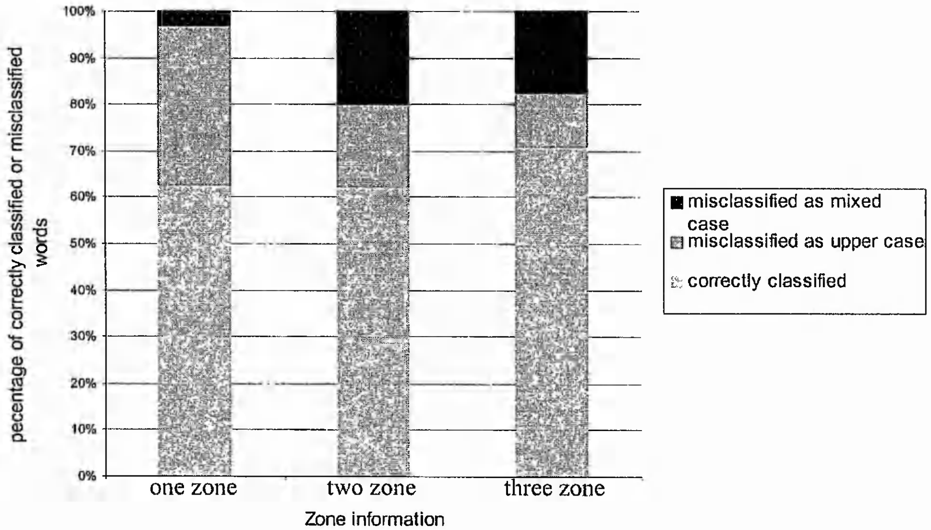



**Figure B-19:** Percentage of lower words, which are correctly or incorrectly classified (lower/upper/mixed data set) using 36 features.

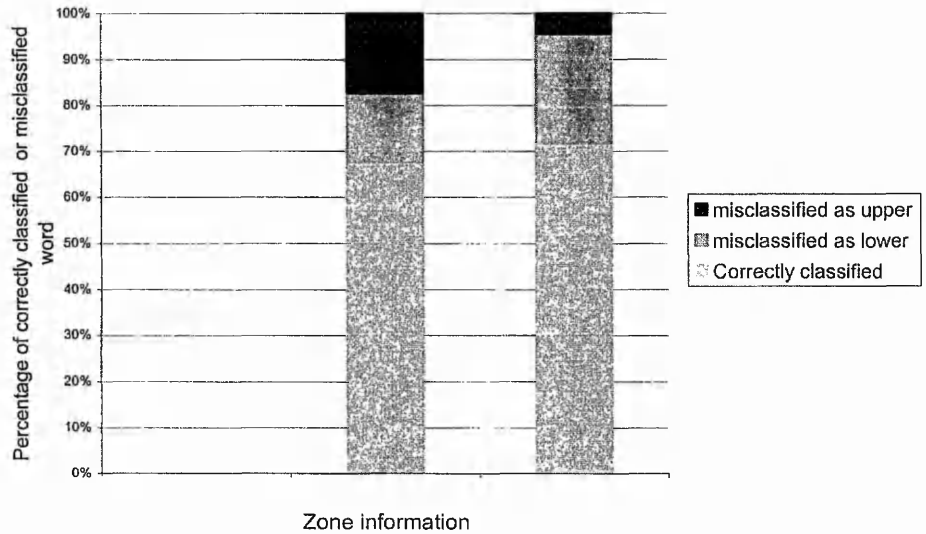



**Figure B-20:** Percentage of mixed words, which are correctly or incorrectly classified (lower/upper/mixed data set) using 36 features.

Figure B-19 shows that 35.00%, 18.00% and 12.00% of one, two and three zone lower word images are misclassified as upper case word images whilst 3.00%, 20.00% and 17.00% of one, two and three zone lower word images are misclassified as mixed case word images respectively. These figures show that one zone lower case words are hardly ever misclassified as mixed case words. The number of two and three zone lower words which are misclassified as mixed case words is roughly the same.

Figure B-20 shows that 15.00% and 24.00% of two and three zone mixed case word images are misclassified as lower case word images whilst 17.00% and 4.00% of two and three zone mixed word images are misclassified as upper case word images respectively.

### B.4.2 Analysis of zoning information using selected features

Figures B-21 and B-22 show the affect of zoning on this classification method.

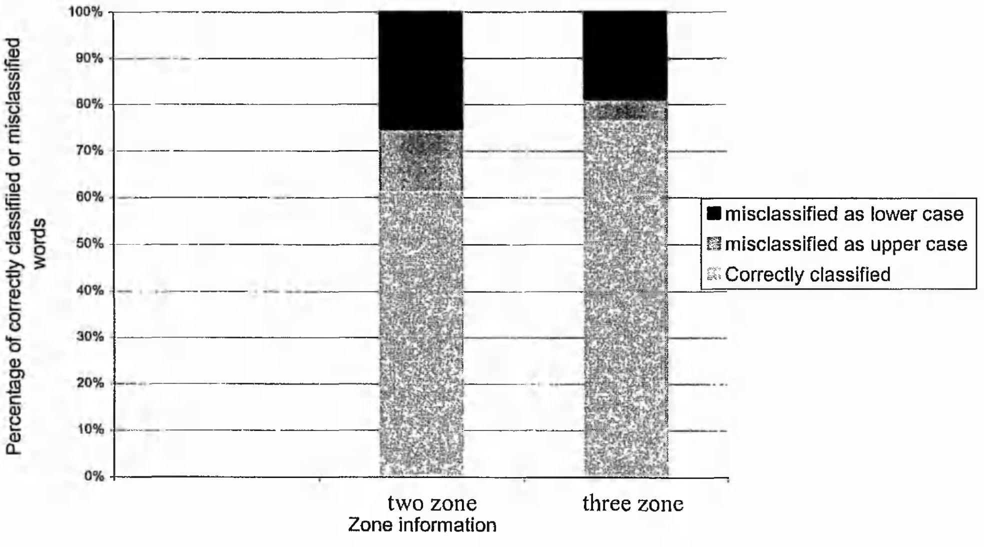



**Figure B-21:** Percentage of mixed words, which are correctly or incorrectly classified (lower/upper/mixed data set) using 25 selected features.

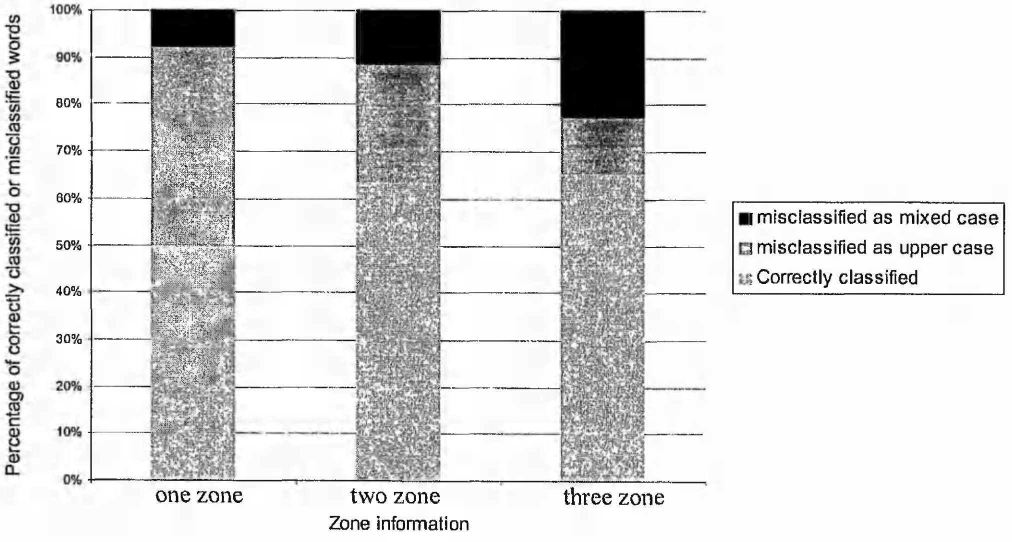



**Figure B-22:** Percentage of lower words, which are correctly or incorrectly (lower/upper/mixed data set) using 25 selected features.

Figure B-21 shows that 62.00% and 77.00% of two and three zone mixed case words are correctly classified and 13.00% and 4.00% of two and three mixed case words are misclassified as upper and 25.00% and 19.00% of mixed case words are misclassified as lower case word images.

Figure B-22 shows that 50.00%, 64.00% and 66.00% of one, two and three zone mixed case word images are correctly classified and 42.00%, 25.00% and 12.00% of one, two and three zone lower case words misclassified as upper case and 8.00%, 11.00% and 22.00% of one, two and three zone lower case words are misclassified as mixed case word images.

# Appendix C. Published papers

## Classification of Off-line Hand-written Words into Upper and Lower Cases

M.Ebadian Dehkordi, N.Sherkat and R.Whitrow

Department of Computing, Nottingham Trent University, Burton Street,
Nottingham.NG14BU
E-mail: {mand,ns,rjw}@doc.ntu.ac.uk

**Key words:** Features extraction, Principal component, Pre-classification of words, Upper and lower case classification, and Contours extraction.

### Abstract

This paper presents an efficient technique for classification of off-line hand-written words into upper and lower case using principal components (PC). The technique consists of two phases. For each word, in feature extraction phase, first the boundary points of the word are extracted, then twenty-six features including global, local, region and dominants features are extracted using the contour information. In the classification phase, a discriminate function based on the PC, adapted by our system, is introduced to integrate the extracted features and classify words into upper and lower case.

Experimental results show that the system achieves an 83% correct word case classification for about 2240 test words randomly selected from a 3226 data set obtained from 12 writers.

## 1. Introduction

Handwriting recognition has been the subject of intensive research for many years. However, despite effort by many researchers, the problem of handwriting recognition is far from solved. The greatest difficulty is due to large variations of shapes resulting from the writing style [1][2].

Among the various types of tasks in handwriting recognition, a pre-classification of words to upper and lower case would provide a useful means of reducing ambiguity. However there are no specific references to research in classifying hand-written words into upper and lower case in the literature. This research therefore focuses on the problem of classifying words to upper and lower case as a prior stage to the recognition stage.

The paper is organised as follows: Section 2 describes the feature selection scheme; Section 3 details how the principal components, as a discriminant function, is applied to the classification problem; Section 4 shows the experimental results obtained from different handwriting sample, conclusion and discussion is on section 5.

## 2. Feature extractions

In this stage a number of useful features are introduced which are based on the outer contour of the hand-written word.

For each word first the boundary information of the word are extracted [3], then twenty six features including a family of global, local, region and dominants features [4] are extracted using the contour information.

A hand-written word can be described as a sequence of separate loop contours

$$W = \{C_i | C_i \cap C_j = \phi,\ i \neq j,\ i = 1,2,\ldots,N\}.$$

Each loop contour $C_i$ is a sequence of consecutive points on the x-y plane:

$C_i = \{p_i |\ i = 1,2,\ldots,M_i\}$,

Where $p_1$ and $p_{M_i}$ are the end points of $i^{th}$ loop contour.

The contour-based features used in the system are mainly based on:

(a) *The chain coding scheme from the eight primitive directions given by Freeman encoding* [5].

Each loop contour $C_i$ can be represented by a chain code sequence

$D_i = \{d_j \mid j = 1,2,\ldots,M_i - 1\}$, and

$$D = \bigcup_{i=1}^{N} D_i$$

(b) *Consecutive exterior angles and contour angles formed by pairs of arrows along the segmented region of the word.*

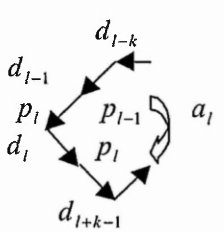



Figure 1

Figure (1) shows the exterior angle $a_l$ at point $p_l$ formed by the pair of vectors $d_l$ and $d_{l-1}$, and is located on the left-hand side of the vectors. The sequences of exterior angles in a loop contour $C_i$, are calculated as:

$A_i = \{a_i | \ i = 2,3,\ldots, M_i - 1\}$ .

(c) Dominant *points*.

Dominant points refer to points of the following types:

1) End points of the segmented regions of each individual loop contour.

2) Points corresponding to local extreme of curvatures of each individual loop contour.

3) Midpoints between two consecutive points of type (1) or (2).

## 3. Classification

In phase I, 26 potentially important features were proposed to be extracted. Processing such a large number of features leads to some problems. These problems are as flow:

1. Speed: Any classification techniques dealing with a large number of variables is slow and time consuming.

2. Correlation: There can be substantial correlation between features. The more features present, the higher the probability of significant interdependencies

All above reasons make for using the principal component. This technique allows extraction of useful information present in a large set of features by means of as few new features as possible [6].

All together, in the classification phase, a discrimination function based on the PC is introduced to integrate the extracted features and classify words into upper and lower case.

## 4. Experimental results:

The classification system has been trained on a data set of word images produced by 12 writers and tested on data set of word images produced by 12 writers (word images different from training set).

Table 1 shows an experiment. The first and third columns show the samples (writers) that are used as a test data set for the lower and upper case respectively. Contents of the second and fourth column are the classification rates in lower and upper case words. Experimental results are as follows:

| Uppercase Sample | %correct | Lowercase Sample | %correct |
|---|---|---|---|
| Writer1 | 83 | Writer1 | 88 |
| Writer2 | 91 | Writer2 | 83 |
| Writer3 | 89 | Writer3 | 59 |
| Writer4 | 83 | Writer4 | 87 |
| Writer5 | 91 | Writer5 | 84 |
| Writer6 | 65 | Writer6 | 83 |
| Writer7 | 91 | Writer7 | 85 |
| Writer8 | 98 | Writer8 | 52 |
| Writer9 | 80 | Writer9 | 68 |
| Writer10 | 93 | Writer10 | 75 |
| Writer11 | 94 | Writer11 | 88 |
| Writer12 | 92 | Writer12 | 74 |

Table 1

## 5. Discussion and conclusion

This paper describes an efficient method for classification of off-line hand-written words into the upper and lower case. A discriminant function based on the PC, adapted by our system, is introduced to integrate the extracted features and classify words into upper and lower case. The experimental results shows that all features used in this system are necessary for reliable classification. Using different writer samples has subjectively validated the system. This indicates that the system is capable of classifying words about 83% accuracy.

The presented approach was restricted to the upper and lower case classification. Therefore the research is ongoing to build up more comprehensive knowledge of the type of hand writing cases which need to be classified, and the characteristics and features which are necessary for their reliable classification.

## References:

[1] B.Yanikoglu and P.A. Sandon, Segmentation of off-line cursive hand writing using linear programing. Pattern recognition, vol.31, no.12,pp.1825-1833, 1998.

[2] L. Schomaker, G. Aabbink and S. Selen. Writer and Writing-style Classification in Recognition of Online Handwriting. Proceeding of the European Workshop on Handwriting Analysis and Recognition: A European Perspective, 12-13 July, 1994, London: The Institution of Electrical Engineers, Digest Number 1994/123, (ISSN 0963-3308).

[3] C H Chien and J K Aggarwal.Model construction and shape recognition from occluding contours. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 11(4): 372-389,1989.

[4] Li and N.S. Hall. Corner detection and shape classification of on-line hand-printed Kanji strokes. *Pattern recognition*, vol.26, no.9, pp.1315-1334, 1993.

[5]H Freeman.On the encoding of arbitrary geometric configuration. *IRE Transactions on Electronic Computers*, EC-10(2):260-268,1961

[6] R.C. Gonzalez and R.E. Wood. Digital Image Processing. Addison Wesley, 1993.

# A Principal Component Approach to Classification of Hand-written Words

M. Ebadian Dehkordi, N. Sherkat, R. J. Withrow
E-mail: {mand, ns, rjw}@doc.ntu.ac.uk
The Notthingham Trent University
Burton Street
Nottingham, NG1 4BU

## Abstract

*This paper presents an efficient technique for classification of off-line hand-written words into upper and lower case using principal components (PC). The technique consists of two phases. For each word, in feature extraction phase, first the boundary points of the word are extracted, then twenty-six features including global, local, region and dominants features are extracted using the contour information. In the classification phase, a discriminante function based on the PC, adapted by our system, is introduced to integrate the extracted features and classify words into upper and lower case.*

*Experimental results show that the system achieves 83% correct word case classification for about 2240 test words randomly selected from a 3226 data set obtained from 12 writers.*

## 1. Introduction

Handwriting recognition has been the subject of intensive research for many years. However, despite effort by many researchers, the problem of handwriting recognition is far from solved. The greatest difficulty is due to large variations of shapes resulting from the writing style [1][2].

Among the various types of tasks in handwriting recognition, a pre-classification of words to upper and lower case would provide a useful means of reducing ambiguity. However there are no specific references to research in classifying hand-written words into upper and lower case in the literature. This research therefore focuses on the problem of classifying words to upper and lower case as a prior stage to the recognition stage.

The paper is organised as follows: Section 2 describes the feature selection scheme and procedures used to compute the features; Section 3 details how the principal components, as a discriminante function, is applied to the classification problem; Section 4 shows the experimental results obtained from different handwriting samples.

## 2. Feature Extraction

In this section a number of useful features are introduced which are base on the outer contour of the hand-written word.

For each word first the boundary information of the word are extracted [3], then twenty six features including a family of global, local, region and dominant features [4][5] are extracted using the contour information.

A hand-written word can be described as a sequence of disjointed loop contours

W= $\left\{ C_i | C_i \cap C_j = \phi,\ i \neq j,\ i = 1,2,\ldots,N \right\}$.

Each loop contour $C_i$ is a sequence of consecutive points on the x-y plane:

$C_i = \left\{ p_i |\ i = 1,2,\ldots, M_i \right\}$,

Where $p_1$ and $p_{M_i}$ are the end points of $i^{th}$ loop contour.

The contour-based features used in our system are mainly based on:

(a) The chain coding from the eight primitive directions given by Freeman encoding [6].

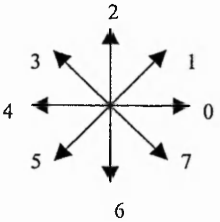



**Figure 1 chain code directions**

Figure 1 refers to the eight primitive directions and represents the writing direction from a start point to end point by following the upper outer contour of the word. Each loop contour $C_i$ can be represented by a chain code

Sequence $D_i = \{d_j \mid j = 1,2,\ldots,M_i - 1\}$, and

$$D = \bigcup_{i=1}^{N} D_i$$

(b) Consecutive exterior angles and contour angles formed by pairs of vectors along the segmented region of the word.

Figure 2 shows the exterior angle $a_l$ at point $p_l$ formed by a pair of vectors $d_l$ and $d_{l-1}$, and is located on the

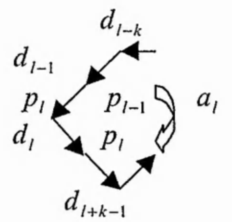



**Figure 2 Exterior angle $a_l$ at $p_l$**

left-hand side of the vectors. The value of $a_l$ can be obtained easily using a lookup table (Table 2.1). The sequences of exterior angles in a loop contour, $C_i$, is calculated as:

$$A_i = \{a_j \mid j = 2,3,\ldots,M_i - 1\}.$$

(c) Dominant points.

Dominant points refer to points of the following types:

1) End points of the segmented regions of each individual loop contour.
2) Points corresponding to local extreme of curvatures of each individual loop contour.
3) Midpoints between two consecutive points of type (1) or (2)[7].

| $(d_{l-1} - d_l)\bmod 8$ | $a_l$ |
|---|---|
| 0 | 180 |
| 1 | 135 |
| 2 | 90 |
| 3 | 45 |
| 4 | 315 |
| 5 | 270 |
| 6 | 225 |

**Table 1 $a_l$ as a function of $(d_{l-1} - d_l)$**

## 3. Classification using Principal Components

In section 2, 26 potentially important features were proposed to be extracted. Processing such a large number of features leads to some problems. These problems are as flow:

1. Speed: Any classification technique dealing with a large number of variables is slow and time consuming.
2. Correlation: There can be substantial correlation between features. The more features present, the higher the probability of significant interdependencies

All above reasons make for using the principal component approach, a technique for extracting the useful information present in a large set of features using as few new features as possible [8].

### 3.1 Principal Components

One of the most common methods of data reduction is that of *principal components.*

Each principal component y, is a fixed linear transform $\Phi$, of the features vector F:

$$y = \Phi F = \phi_1 f_1 + \phi_2 f_2 + \ldots\ldots$$

In our case, F = $(f_0, f_1, \ldots, f_{25})$ where $f_i$ are the features are used in the system.

In mathematical terms $\Phi$ is defined in such a way that the variance $\Phi$F is maximised relative to the universe of possible feature vectors subject to the length of $\Phi$ being fixed. The second PC can be defined in the same manner to capture maximum variation from the collection but subject to the restriction that it is uncorrelated with the first PC.

An optimal solution to the computation of $\Phi$ is given in terms of the eigenvectors of the features covariance matrix W.

The covariance matrix W for our training set is a 26 by 26 symmetric matrix whose diagonal contains the variances of each of the 26 extracted features and whose off-diagonal area contains their covariance. Each of W's elements is computed using the following Equation.

$$w_{ij} = w_{ji} = \frac{1}{n}\sum_{k=1}^{n}(f_{ki} - \mu_i)(f_{kj} - \mu_j) \text{ where}$$

$$\mu_j = \frac{1}{n}\sum_{i=1}^{n} f_{ij}$$

and $f_{ij}$ stands for the vector extracted feature $j$ in sample $i$ of collection of n sample that comprises the training set.

### 3.2 Discriminate function

As previously mentioned principal components capture the maximum variance from a collection of sample cases relative to all possible collections with many variables. This can be a disadvantage in case that the collection is composed of samples from different classes. The data is treated as a single group.

Any categories inherent in the collection are ignored when principal components are computed. This means that the principal component extracts useless information (i.e variation within classes) along with useful information (i.e variation between classes).

One way to tackle this problem is to consider the mean vector of all feature vectors within a class as a representative for that class, in our case $\bar{f}_i$, and $\bar{g}_i$ respectively for upper and lower case. Then compute the principal component base on the class means.

$\bar{f}_i = \frac{1}{26}\sum_j f_{ij}$ (Mean of $i^{th}$ features for lowercase words sample) and

$\bar{g}_i = \frac{1}{26}\sum_j g_{ij}$ (Mean of $i^{th}$ features for uppercase words sample)

In this way variation between classes are ignored. Our experiments show that results based on group centres are poor. One reason behind this might be that the new computed principal component, that is supposed to optimally discriminate between classes, is based on the difference between classes (centres) not the distribution within the classes. It means that one should take advantage of shape of the within classes distribution to compute the new variable that is very effective in discrimination between the classes.

For this reason it is assumed that the training set consists of samples from each of several different classes. Based on principal components the **V** matrix will be the eigenvectors of the total variance (the variance within classes **W**, and between classes **B**). Therefore, instead of basing the principal component on **V**, we compute the principal component based on $W^{-1}B$ matrix. This means that the **V** matrix will be the eigen vectors of $W^{-1}B$ matrix. In this case, the eigen values are no longer the variances of the discriminant functions.

Therefore to find the new values, one must expilicity compute the variance of each discriminant function, then divide the columns of **V** by the square root of that quantity. It can be proved that for a particular column of V, $v_i$ the variance of that discriminant function within each class is given by the quadratic form $\tau = v_i^{-1} w v_i$.

## 4. Experimental Result and Analysis

The handwriting samples used in the experiments were selected from a database containing 150 words from 12 writers. The words in the database are written by numerous writers: one written all in lowercase, and one written all in upper case, without any other constraints on the writing style. In the following experiments, $LW_i$ and $UW_i$ refer to samples written by $i^{th}$ writer.

Table 4.1 shows an experiment. The first and third columns show the samples (writers) that are used as a test data set for the lower and upper case respectively. Contents of the second and fourth column are the classification rates in lower and upper case words. The classification system has been trained on a data set of 1000 word images produced by 12 writers and tested on data set of 2240 word images produced by 12 writers (word images different from training set).

| Uppercase Sample | %correct | Lowercase Sample | %correct |
|---|---|---|---|
| LW1 | 83 | UW1 | 88 |
| LW2 | 91 | UW2 | 83 |
| LW3 | 89 | UW3 | 59 |
| LW4 | 83 | UW4 | 87 |
| LW5 | 91 | UW5 | 84 |
| LW6 | 65 | UW6 | 83 |
| LW7 | 91 | UW7 | 85 |
| LW8 | 98 | UW8 | 52 |
| LW9 | 80 | UW9 | 68 |
| LW10 | 93 | UW10 | 75 |
| LW11 | 94 | UW11 | 88 |
| LW12 | 92 | UW12 | 74 |

**Table 2 Classification results**

## 5. Discussion and conclusion

This paper describes an efficient method for classification of off-line hand-written words into upper and lower case. A discriminant function based on the PC, adapted by our system, is introduced to integrate the extracted features and classify words into upper and lower case. The experimental results show that all features used in this system are necessary for reliable classification. Using different writer samples has objectively validated the system. This indicates that the system is capable of classifying words to about 83% accuracy.

The presented approach is restricted to the upper and lower case classification. Therefore the research is ongoing to build up more comprehensive knowledge of the type of hand writing cases which need to be classified, and the characteristics and features which are necessary for their reliable classification. This is planned to extend the current development to include classification of the mixed case cursive words. Providing a means of pre-classifying word images into upper, lower and mixed case is expected to provide a significant contribution as currently most of the reported algorithms simply assume this pre-classification.

## 6. References


[1] B.Yanikoglu and P.A. Sandon, Segmentation of off-line cursive hand writing using linear programming. Pattern recognition, vol.31, no.12, pp.1825-1833, 1998.

[2] L. Schomaker, G. Aabbink and S. Selen. Writer and Writing-style Classification in Recognition of Online Handwriting. Proceeding of the European Workshop on Handwriting Analysis and Recognition: A European Perspective, 12-13 July, 1994, London: The Institution of Electrical Engineers, Digest Number 1994/123, (ISSN 0963-3308).

[3] C. H. Chien and J K Aggarwal. Model construction and shape recognition from occluding contours. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 11(4): 372-389,1989.

[4] R. K. Powalka, N. Sherkat, R J Whitrow, Zoning Invariant Wholistic Recognizer for Hybrid Recognition of Handwriting Procs. Of ICDAR'95, Third International Conference on Document Analysis and Recognition, published by IEEE Computer Society Press, Los Alamitos, California, Eds Mary Kavanaugh and penny Storms, pp 64-67, held in Montreal, Canada, August 14-16 1995.

[5] G. H. Reza, A Henning, N Sherkat and R J Withrow, Recognition of Facsimile Documents using a Database Of Robust Features, Fourth International Conference on Document Analysis and Recognition ICDAR'97, pp.4444-4448, Ulm, Germany, 18 August 1997, ISBN 0-8186-7898-4.

[6] H. Freeman. On the encoding of arbitrary geometric configuration. *IRE Transactions on Electronic Computers*, EC-10(2):260-268,1961

[7] Li and N. S. Hall. Corner detection and shape classification of on-line hand-printed Kanji strokes. *Pattern recognition*, vol.26, no.9, pp.1315-1334, 1993.

[8] R.C. Gonzalez and R.E. Wood. Digital Image Processing. Addison Wesley, 1993.

# Case Classification of Off-line Hand-written Words Prior To Recognition

Mandana Ebadian Dehkordi, Nasser Sherkat and Tony Allen

IRIS, Department of Computing, The Nottingham Trent University
Burton Street, Nottingham, NG1 4BU,U.K.
E-mail: {mand,ns,tja}@doc.ntu.ac.uk

**Abstract.** Pre-classification of words to upper, lower and mixed cases would provide a useful means of reducing word ambiguity. If it were possible to classify the case of a word image prior to recognition, then the size of the lexicon used for any individual word recognition could be significantly reduced. This paper presents an efficient technique for classification of off-line hand-written words into upper, lower and mixed case using principal components (PC). The technique consists of two phases. In the feature extraction phase, the boundary points of each word are first determined, then thirty-six features are extracted using this contour information. In the classification phase, a discriminant function is applied to integrate the extracted features and classify words into upper, lower and mixed case. Experimental results show promising results. The system achieves 93.44%, 97.38%, 74.88% correct word case classification for upper and lower, upper and mixed, and lower and mixed case words respectively.

## 1 Introduction

Although research in recognising hand-written characters and numerals has reached a reasonable stage of development, recognition of unconstrained cursive handwriting has proven to be much more difficult. The greatest difficulty is due to the large variations in shapes that result from the different writing styles [1]. Previous research has shown that writing style can vary significantly with geographical location, cultural background, age, sex and so forth [2].

It is hypothesised that one way of helping cursive script recognition systems would be to detect writing style prior to the recognition stage. As an example, pre-classification of words into upper, lower and mixed case would provide a useful means of reducing this style ambiguity. However there are no reports of significant research in classifying handwriting into upper, lower and mixed case in the literature. This research therefore focuses on the problem of classifying words to upper, lower and mixed case as a prior stage to the recognition stage.

The paper is organised as follows: Section 2 describes the feature selection scheme and procedures used to compute the features; Section 3 details how the principal components, as a discriminante function, is applied to the classification problem; Section 4 shows the experimental results and analysis. Section 5 gives a brief summary and conclusion.

## 2 Feature Extraction

In this section a number of useful features are introduced which are based on the outer contour of the hand-written word [3]. A hand-written word can be described as a sequence of separate loop contours (1).

$$W = \{C_i \mid C_i \cap C_j = \phi, i \neq j, i = 1,2,...,N\}. \quad (1)$$

Each loop contour $C_i$ (2) is a sequence of consecutive points on the x-y plane:

$$C_i = \{p_i \mid i = 1,2,...,M_i \ \& \ p_i \text{ are consecutive points}\}. \quad (2)$$

Where $p_1$ and $p_{M_i}$ are the end points of $i^{th}$ loop contour.

The contour-based features used in the system are mainly based on:

(a) The chain coding scheme from the eight primitive directions given by Freeman encoding [4].
Each loop contour $C_i$ can be represented by a chain code sequence (3).

$$D_i = \{d_j \mid j = 1,2,...,M_i - 1\}, \text{ and } D = \bigcup_{i=1}^{N} D_i . \quad (3)$$

(b) Consecutive exterior angles and contour angles formed by pairs of arrows along the segmented region of the word.

Figure 1 shows the exterior angle $a_l$ at point $p_l$ (4) formed by the pair of vectors $d_l$ and $d_{l-1}$, and located on the left-hand side of the vectors. The sequences of exterior angles in a loop contour $C_i$, are calculated as:

$$A_i = \{a_i \mid i = 2,3,...,M_i - 1\}. \quad (4)$$

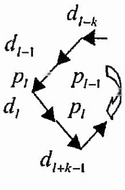



**Fig 1.** Exterior angle $a_l$ at point $p_l$

*(c) Dominant points.*

Dominant points refer to points of the following types:

1. End points of the segmented regions of each individual loop contour.
2. Points corresponding to local extreme of curvatures of each individual loop contour.
3. Midpoints between two consecutive points of above types.

The above information is used to introduce features such as contour-based features, global features, region-based features, windows based features, features based on moments and features based on zero crossing. These features were used for classification between upper and lower case words in our previously reported work [5][6].

In addition to the above features, in this paper we introduce additional features to help distinguish between the more difficult cases of mixed and lower case. These features are group-based features and features based on the horizontal histogram, which are described as follows:

### 2.1 Group Definition

To avoid using any segmentation technique, which may lead to errors, group-based features are introduced. A group can be described as a set of pixels in a word image, which contain one outer loop contour (4).

$$W = \{G_i \mid G_i \cap G_j = \phi, i \neq j, i = 1,2,....,N, j = 1,2,....,N\} \text{ or } W = \bigcup_{i=1}^{N} G_i \qquad (4)$$

where $G_i = \{p_i \mid i = 1,2,...,N_i \;\&\; p_i \text{ have one outer loop contour}\}$,

$p_i = \{(x_i, y_i) \mid i = 1,2,...N_i\}$

and $N$ is the number of groups in a word and $N_i$ is number of pixels in $i^{th}$ group of each word.

The group features used in our system are mainly based on:

(a) Zoning information.

The zoning lines of the word image are the four lines that partition the word into three disjoint horizontal slices or zones. The width of upper and lower zone is 25% of the word height and width of middle zone is 50% of the word height.

(b) Bounding box of each group.

A bounding box is a rectangular shape constructed of four points $P_{mima,i}$, $P_{mimi,i}$, $P_{mama.i}$ and $P_{mami,i}$ (Figure2) denote the intersections between four lines, two horizontal line passing through the $Y_{\min,i}$, $Y_{\max,i}$ positions and two vertical lines passing through the positions. $Y_{\min,i}$, $Y_{\max,i}$ denote the minimum and maximum value of $y_i$ and $X_{\min,i}$, $X_{\max,i}$ denote the minimum and maximum value of $x_i$ for each pixel in the $i^{th}$ group respectively.

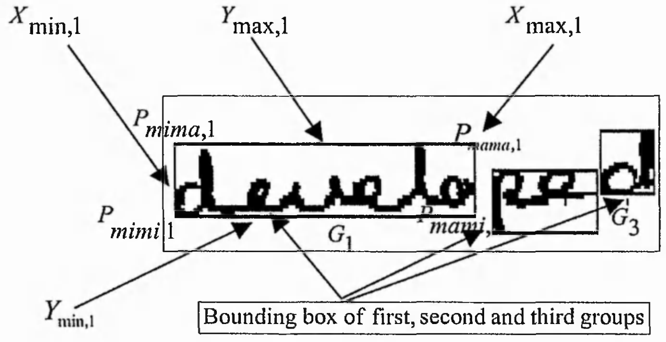



**Fig 2.** Groups and their bounding box.

### 2.1.1 Group-based Features

The following group-based features are used in our system based on the above definition of groups.

(1) Number of groups in each word (N). Total number of groups in a word.

Since the first few letters in a word hold more reliable information, only features present in the first three groups in a word images are considered [7]. Furthermore our experimental result show that increasing the number of groups is not beneficial and can lead to confusion. Therefore the following features are extracted from first three groups of each word.

(2) Ratios of distance between upper bounding line and upper zone line to distance between lower and upper zone line for the first three groups of the word (5) (Figure 3).

$$RDUU_i = \frac{Y_U - Y_{\max,i}}{Y_U - Y_L}, \quad i = \{1,2,3\} \tag{5}$$

(3) Ratios of distance between lower bounding line and lower zone line to distance between lower and upper zone line for the first three groups of the word (6) (Figure 3).

$$RDLL_i = \frac{Y_L - Y_{\min,i}}{Y_U - Y_L}, \{i = 1,2,3\} \tag{6}$$

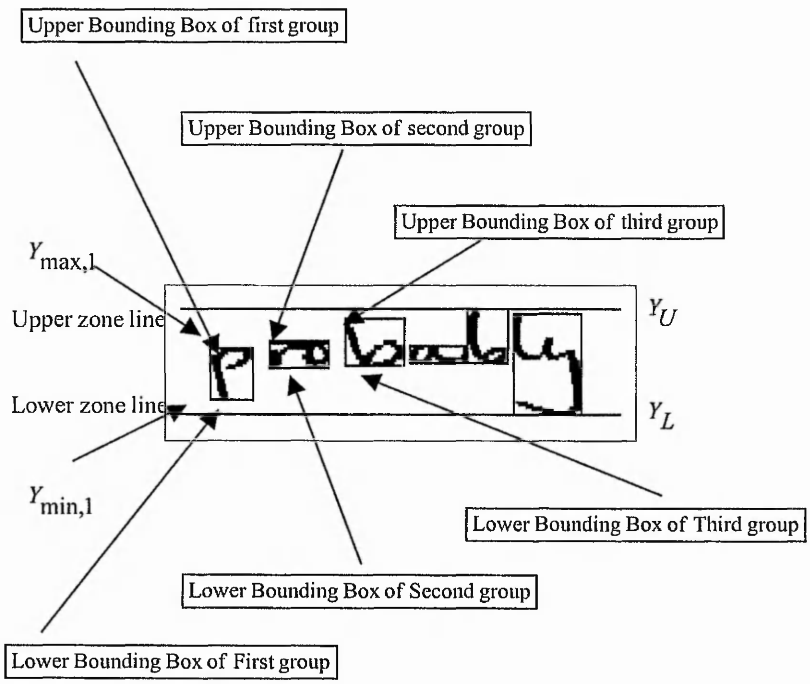



**Fig 3.** Illustration of group-based features.

### 2.1.2 Horizontal-based Histogram Features

Different characteristics of the horizontal pixel histogram are examined (Figure 4). The mean value of the columns on the horizontal histogram are calculated by (7).

$$m = \frac{\sum_{i=1}^{n} col_i}{n} \quad . \tag{7}$$

Where $col_i$ is number of black pixels in $i^{th}$ column of horizontal histogram and $n$ is number of columns in histogram.

*(1)* Spread or first moment of the histograms (8):

$$\mathrm{FMH} = \frac{\sum_{i=1}^{n} |col_i - m|}{mn} \quad . \tag{8}$$

*(2)* The distance of the average height of columns (9):

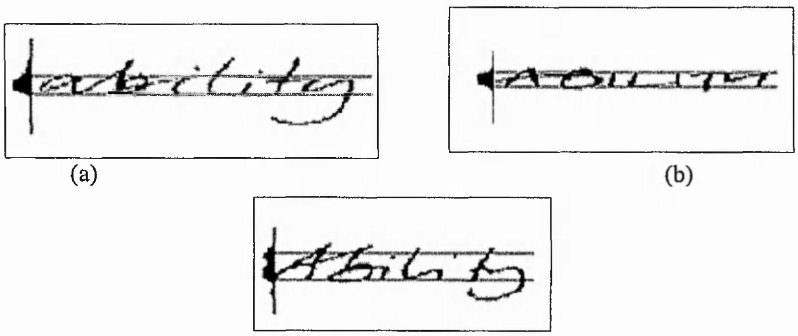
$$AH = \frac{K_2 - K_1}{n} \quad . \tag{9}$$

Where

$$K_1 = \min_i \left\{ k_i : k_i = \frac{col_i + col_{i+1}}{2} > m, i = 1,2,\ldots,n \right\}. \quad (10)$$

$$K_2 = \min_i \left\{ k_i : k_i = \frac{col_i + col_{i-1}}{2} > m, i = 1,2,\ldots,n \right\}. \quad (11)$$

*(3) Ratio of number of black pixels in upper zone to number of black pixels in all three zones of a word.*



(a) (b) (c)

**Fig 4.** Horizontal histograms of (a) An uppercase word (b) Lowercase word (c) Mixed case word all written by the same writer

## 3 Classification

In this paper a dicriminant function based on principal component is used in classification method. One of the most common methods of data reduction is that of *principal components (PC)* [8]. Principal components attempt to eliminate irrelevant information by transforming the original set into a new set of variables with little loss of information.

### 3.1 Principal Components

The covariance matrix W for our training set is a 36 by 36 symmetric matrix whose diagonal contains the variances of each of the 36 extracted features $(f_1, f_2, \ldots, f_{36})$ and whose off-diagonal area contains their covariance. Each of the elements in matrix W is computed using the Equation (12).

$$w_{ij} = w_{ji} = \frac{1}{n} \sum_{k=1}^{n} \left(f_{ki} - \mu_i\right)\left(f_{kj} - \mu_j\right). \quad (12)$$

Where

$$\mu_j = \frac{1}{n} \sum_{i=1}^{n} f_{ij}.$$

and $f_{ij}$ stands for the vector extracted feature $j$ in sample $i$ of collection of n sample that comprises the training set.

The principal components obtained from the correlation matrix W of a set of features shows that the 36 features are weakly correlated. The eigen values are slowly decreasing. For example the twenty-four features cover 90% of the variability. Figure 5 shows the variation of variance relative to the number of principal components.

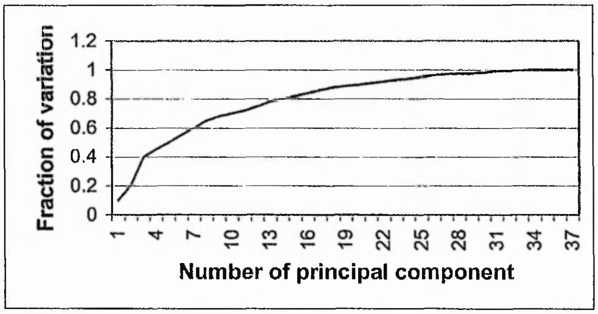



**Fig 5.** Variation of variance relative to the number of principal components

### 3.2 Discriminant Function

It mentioned in our previous work [6], that as discrimination between classes based on the difference between the mean of feature for each class is poor therefore distribution within classes should be considered.

It is assumed that the training set consists of samples from each of three different classes. Based on principal components the **V** matrix will be the eigen vectors of the total variance (the variance within classes **W**, and between classes **B**). Therefore, instead of basing the principal component on **V**, we compute the principal component based on $W^{-1}B$ matrix. This means that the **V** matrix will be the eigen vectors of $W^{-1}B$ matrix. In this case, the eigen values are no longer the variances of the discriminant functions.

Therefore to find the new values, one must explicitly compute the variance of each discriminant function, then divide the columns of **V** by the square root of that quantity. In this work the decision for assigning each word to a particular class is based on measuring the Euclidean distance between its score (calculated by discriminant function) and the centroids (mean of scores) of each training set.

## 4 Experimental Result and Analysis

Previous work [9] had indicated the need for careful choice of sample words to allow a good representation of a much large vocabulary without becoming hopeless unwieldy. Kassel [10] has discussed design aspects of such data sets and sample words used in this research were designed from this work. Therefore this technique was applied on our existing data set, scanned images obtained from 9 writers containing 150 words with 200x100-dpi resolution. Each writer has written each word in all lower, all upper and all mixed case without any other constraints on the writing style. In Table 1 the first column shows the case categories and the second column is the result of classification rates in lower, upper and mixed case words. This table shows the classification results achieved on 1667 test words randomly selected from a 2452 data set of each category. The training set consists of 750 words, which are not in the test set of each category.

| Case categories | %Correct Classification Result |
|---|---|
| Upper and lower | 93.44 |
| Upper and mixed | 97.38 |
| Lower and mixed | 74.88 |

**Table 1.** Experimental result

It can be seen from table 1 that the highest classification rate is between the upper and mixed case words. This is due to the fact that virtually all of the upper case words are midzone only whilst the converse is true for their mixed case equivalents. The reduced classification between the upper and lower case words is then due to the fact that a proportion of the lower case words are midzone only (i.e. "are", "can", "an", " now" and etc). These lower words are then incorrectly classified as upper case.

The lower classification rate between the lower and mixed case words is due to a variety of factors. These include; one group only words, lower case words with an ascender in the first character position and the poor quality of writing within the mixed case data set.

The first factor, one-group only words, mainly affects short words such as "is", "be" and "an" etc. or specific writers who use totally cursive handwriting. Such one-group lower case words are usually incorrectly classified as mixed case as $RDUU_1$ and $RDLL_1$ are zero for such words. The horizontal histogram features were an attempt to overcome this problem but it only worked for a small number of cases. Although some uppercase words contain characters like 'L','T','E', which distort the horizontal histogram, experiments show that horizontal histograms of mixed and lower case words are sharper than upper case words because of the presence of ascenders or descenders. In other words, unlike the upper case, the pixel density of lower case words in the middle zone is more than in the upper and lower zones because of ascenders and descenders. This property was used to improve the classification results on purely cursive handwriting.

Lower case words with ascenders in the first character position suffer from the opposite problem in that they tend to be misclassified as mixed case. This is because of the small distance between them and their mixed case equivalent.

The final factor, poor quality of writing, is probably due to the fact that the writers where required to artificially produce sentences with each word written as mixed case. This unnatural writing style tended to produce a poorer quality of writing, which is affected the classification rate. Indeed in some instances the human reader could not tell the difference between the lower and mixed case equivalents ("can" and "Can" or "so" and "So" and etc).

## 5 Discussion and Conclusion

This paper describes a method for the classification of off-line hand-written words into upper, lower and mixed case. A discriminant function based on the PC, adapted by our system, is introduced to integrate the extracted features and classify words into upper, lower and mixed case. The experimental results show that all features used in this system are necessary for reliable classification. Using different writer samples has subjectively validated the system. The results indicate that the system is capable of classifying words to upper, lower and mixed cases.

The presented approach was restricted to the upper, lower and mixed case words classification. Therefore research is ongoing to build up a more comprehensive knowledge of the types of hand writing style and the characteristics and features which are necessary for their reliable classification. Our observation of the adverse effect of unnatural mixed case data collection has prompted investigation into alternative means of collecting data. Determining an effective means of collecting mixed case data forms part of our ongoing research.

## 6 References

[1] L. Schomaker, G. Aabbink and S. Selen. "Writer and Writing-style Classification in Recognition of Online Handwriting". Proceeding of the European Workshop on Handwriting Analysis and Recognition,The Institution of Electrical Engineers, (ISSN 0963-3308) London, July 1994.

[2] Powalka R. K., "An algorithm tool-box for on-line cursive script recognition", PhD Thesis, Dept. of Computing, The Nottingham Trent University, May 1995

[3] C H Chien and J K Aggarwal. "Model construction and shape recognition from occluding contours". IEEE Transactions on Pattern Analysis and Machine Intelligence, 11(4): 372-389,1989.

[4] H Freeman.On the encoding of arbitrary geometric configuration. IRE Transactions on Electronic Computers, EC-10 (2): 260-268,1961

[5] M. Ebadian Dehkordi, N. Sherkat and R. J. Whitrow. "Classification of Off-line Hand-written words into Upper and Lower cases", IEE Colloquium Document Image Processing and Multimedia", 8/1-8/4, London, March 1999.

[6] M. Ebadian Dehkordi, N. Sherkat and R. J Whitrow. "A Principal Component Approach to Classification of Hand-written Words", Fifth International Conference on Document Analysis and Recognition (ICDAR' 99), 781-784, India, September 1999.

[7] Shila X. Zhoa and Sargur N. Srihari.Word Recognition using a Lexicon Constrained by First/ Last Character Decisions, SPIE vol.2422, 98 –104, 1995.

[8] R.C. Gonzalez and R.E. Wood. Digital Image Processing. Addison Wesley, 1993.

[9] M. Jedrzejewski, Automatic Characterization of Handwrting Style, Mphil Thesis, July 1996.

[10] R. H. Kassel, "A comparison of aproaches to on-line handwritten character recognition", Docttoral Dissertation, Department of Electronical Engineering and Computer Science, MIT June 1995.

# Prediction of Handwriting Legibility

Mandana Ebadian Dehkordi, Nasser Sherkat and Tony Allen
E-mail: {maudana.ebadian-dehkordi@ntu.ac.uk}
{ns, tja}@doc.ntu.ac.uk
IRIS, Department of Computing, Nottingham Trent University,
Burton Street, Nottingham, NG1 4BU, UK

## Abstract

*This paper describes an independent handwriting style classifier that has been designed to select the best recognizer for a given style of writing. For this purpose a definition of handwriting legibility has been defined and a method has been implemented that can predict this legibility. The technique consists of two phases. In the feature extraction phase, a set of 16 features is extracted from the image contour. These features have been selected from amongst a set of pre-recognition features as those features that contribute the most (95%) to a discriminant between legible and illegible words. In the classification phase, a Probability Neural Network based on Bayesian decision is introduced to predict the legibility of unknown handwriting using a Parzen method to estimate a class conditional density function from the available training data.*

***Key words:*** *Writing style, legibility of handwriting, Bayesian classification, Parzen model and linear discriminant function.*

## 1. Introduction

Many methods have been developed for handwriting recognition and in general they all attempt to deal with poorly written handwriting [1]. Various algorithms have shown considerable success with certain handwriting styles but most, if not all, cannot maintain their high recognition rates for all styles of handwriting. It is hypothesised that one way of helping cursive script recognition systems would be to detect writing style prior to the recognition stage. In this way the best recogniser could be selected for the style of writing using a prediction of legibility based on the given recogniser's performance. This research therefore focuses on the problem of classifying word images as legible or illegible prior to the recognition stage.

This paper is organised as follows: Section 2 gives a definition of legibility; Section 3 describes the feature extraction process; Section 4 describes a method for feature selection; Section 5 details how a Bayesian decision is applied to the classification problem. Sections 6 and 7 give the experimental results and discussion followed by conclusions and future work in section 8.

## 2. Definition of legibility

Upto now legibility has been defined in human terms. However since the ability of a machine-based recogniser differs significantly from that of a human being, any definition of legibility should be based on the recognition system. Of course similar to that of a human being the definition of legibility is a debatable issue. Considering the novelty of this concept in handwriting recognition at the time of writing no reference to a machine based definition of legibility has been found in the literature.

Our definition of handwritten legibility has therefore been based on our existing recogniser's (HVBC) performance [2]. HVBC is a holistic word level recogniser that uses three features namely, Holes, Vertical bars and Cups. However this definition of legibility can be extended to any available recogniser. Legible words are thus defined as those that are likely to be placed the in the top 10 of the word list with a score of 75 or greater. Illegible words are defined as those that would produce a list containing the word with a score of less than 45 any where in the word list. These thresholds have been arrived at experimentally and merely provide a starting point. They will be reviewed in the light of future experiments to establish their validity.

## 3. Features extraction

During the design process of this classification system thirty-six potentially useful features were first extracted from the contour information of a large number of handwritten word images provided by several different writers [3][4][5].

A hand-written word can be described as a sequence of disjointed loop contours

$$W = \{C_i \mid C_i \cap C_j = \phi, i \neq j, i = 1,2,\ldots,N\}.$$

Each loop contour $C_i$ is a sequence of consecutive points on the x-y plane:
$C_i = \{p_i \mid i = 1,2,\ldots,M_i\}$,

The contour-based features used in our system are based on:

(a) The chain code from the eight primitive directions given by Freeman encoding [6].

(b) Consecutive exterior angles and contour angles formed by pairs of vectors along the segmented region of the word.
(c) Dominant points.

Dominant points refer to points of the following types:
(1) End points of the segmented regions of each individual loop contour.
(2) Points corresponding to local extreme of curvatures of each individual loop contour.
(3) Midpoints between two consecutive points of type (1) or (2).

Using these points the contour of word images can be partitioned into a sequence of convex, concave and plain regions.

## 4. Feature selection

As the number of potential features is large a data reduction method was used to select the best $n$ features for style classification. For this purpose a linear discriminant function was applied on the set of 36 pre-recognition features to select those features that contribute most to a discriminant between legible and illegible words. This discriminant function seeks a set of transformation vectors $a_i$ that maximise $\frac{A^T S_B A}{A^T S_W A}$ where

$$S_B = \sum_{i=1}^{n} \frac{n_i}{n}(m_i - m)(m_i - m)^T, \quad S_W = \sum_{i=1}^{C} \frac{n_i}{n} \mathrm{cov}_i,$$

$$m = \sum_{i=1}^{C} \frac{n_i}{n} m_i \quad \mathrm{cov}_i = \frac{1}{n_i} \sum_{j=1}^{n} (a_j - m_i)(x_j - m_i)^T,$$

$m_i = \frac{1}{n_i} \sum_{j=1}^{n_i} a_j$, $n = \sum_{i=1}^{C} n_i$ and where $C$ is the number of classes, $n_i$ is the number of available samples in each class and $a_i$ are the features introduced in section 3. This set can be found by using the eigenvector equation $S_B A = S_W A \lambda$ or $S_W^{-1} S_B A = A\lambda$ where A is the matrix whose columns are $a_i$ and $\lambda$ is the diagonal matrix of eigenvalues. Features corresponding to the largest elements of the eigenvector are then considered to be the best features for use in the style classification system [7]. Figure 1 shows the selected features corresponding to the largest eigenvector's elements when using a training set. It can easily be seen that just 16 features capture most of the variation between the two classes.

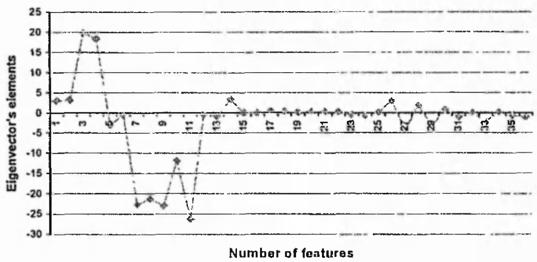



**Figure 1:** The 16 largest eigenvector weights capture 95% of the variability between, legible (Good) and illegible (Bad) handwriting.

The percentage of contribution (CON) of the selected feature sets is the ratio of the sum of eigenvalues that has been selected to all possible eigenvalues:

$$\mathrm{CON} = \frac{\sum_{j=1}^{m} \lambda_j}{\sum_{j=1}^{36} \lambda_j} \times 100\ \% \tag{1}$$

where $m$ is number of selected features.

The selected features, which contribute about 95% of the variation between the classes, are shown in Table 1. For more detail on these features the reader is referred to [4][5].

| Features Number | Feature Description |
|---|---|
| 1 | Average Region Length |
| 2 | Average plain region length |
| 3 | Average concave region length |
| 4 | Average convex region length |
| 5 | Ratio of Original Sharp Angle to the total number of Points |
| 7 | Ratio of critical vertical direction to the total critical chain code |
| 8 | Ratio of critical horizontal direction to the total critical chain code |
| 9 | Ratio of critical diagonal direction to the total critical chain code |
| 10 | An estimate of number of sharp angles in the whole |
| 11 | An estimate of the component length (disjoint contours) or averaged component $(C_i)$ length |
| 14 | Ratio of diagonal direction to the total chain code |
| 26 | First moment feature |
| 27 | Ratio of number of points in middle area to total number of points |
| 28 | Ratio of number of black pixels in the upper zone to number of black pixels in all three zone of a word. |
| 29 | Spread or first moment of the histograms |
| 33 | Ratios of distance between upper bounding box and upper zone to distance between lower and upper zone for the third three groups of the word |

**Table 1.** 16 Selected features

## 5. Classification

A statistical classification method based on a Bayesian rule decision is used to predict the legibility of an unseen word. The basic idea behind the Bayesian estimation is to obtain information about the parameter $\omega$ from observations $x_1, x_2, \ldots x_n$. The probability that a particular pattern $x$ comes from $\omega_i$ is denoted as $p(\omega_i|\underline{x})$ [8] where

$$p(\omega_i|\underline{x}) = \frac{p(\underline{x} \mid \omega_i)p(\omega_i)}{\sum_{j=1}^{C} p(\underline{x} \mid \omega_j)p(\omega_j)}. \tag{2}$$

This equation requires knowledge of the class-conditional density, which can be estimated from the parameters of a model, derived using an available training set. A Parzen model [9] is used to estimate the class density function in this experiment.

### 5.1 Parzen method

The accuracy of the Baysian decision depends on the accuracy with which the underlying class-conditional density is estimated. A Parzen model [12] is a class of smooth and continuous estimators, which becomes more representative of the true class-conditional density as the number of samples increases. The parzen model uses a weight function $W(d)$ which has a maximum value at $d = 0$ and which decreases as the absolute value of $d$ increases. A general formulation of the parzen model is described by:

$$g(\underline{x}) = \frac{1}{n_i} \sum_{j=1}^{n_i} W\left(\frac{\left\|\underline{x} - \underline{x}^j\right\|}{\sigma}\right) \tag{3}$$

where $\underline{x}^j = \left(x^i{}_1, \ldots, x^i_{36}\right)$ are the sample points (extracted features) in the training set. $\sigma$ is the variance of points that surround each sample in the training set, $n_j$ is the number of samples in the training set (in class $w_j$), $W$ is the weight function and $x_l^j$ is the $l^{th}$ feature which is extracted from a word image belonging to the $w_j$ class. In Eq. (4) the Euclidean distance ($D(\underline{x}, \underline{x}^i)$) is first computed and, then divided by a common sigma. A more general density estimator, which assumes a Guassian kernel distribution used in this study is:

$$g(\underline{x}) = \frac{1}{n} \sum_{i=1}^{n} \exp\left(-D\left(\underline{x}, \underline{x}^i\right)\right) \tag{4}$$

where

$$D\left(\underline{x}, \underline{x}^i\right) = \sum_{j=1}^{36} \left(\frac{\underline{x}_j - \underline{x}^i_j}{\sigma_j}\right)^2 \tag{5}$$

is a distance function with different sigma values for each of the 36 extracted features thus

$$p\left(\omega_j|\underline{x}\right) = \frac{1}{\prod_{i=1}^{36} \sqrt{2\pi}\sigma_i} \exp\left\{\sum_{i=1}^{36} \left(\frac{x_{ij} - x_i}{2\sigma_i}\right)^2\right\} \tag{6}$$

In general each Parzen method should have multiple $\sigma_i$. However to simplify the model a special case can be assumed where $\sigma = \sigma_i = \sigma_1 = \sigma_2 = \ldots = \sigma_n$ for all of the weights of function $W$.

### 5.2 Estimation of $\sigma$ based on leave-one-out method

Estimating the range of $\sigma$ is not difficult. For each particular $\sigma$ a set of Parzen density estimators based on the training data set was estimated. The number of correctly classified words produced by each value is then used to judge the efficiency of a particular value of $\sigma$. To estimate an unbiased correct classification rate for each $\sigma$, a leave–one-out method was used. In this method, all of the training data set belonging to each class (legible and illegible) except one is used to train the system and the remaining datum is used for testing. This training and testing using the leave-one-out method was repeated until every datum element in the 2 different classes had been independently tested. This method (leave-one-out) thus gives the legible and illegible bounds of the true performance of the classifier.
The numbers of misclassified words for each $\sigma$ are then counted as an error function. A final value of $\sigma$ is then chosen that minimises the error function (number of misclassifications). The minimisation technique involves two stages. First a global search over a reasonable range is used to find a rough minimum. The range can be determined iteratively such that the error rate is minimised. Then a golden section method [8] is used to refine the estimate.

### 5.3 Probabilistic Neural Network

The non-parametric classifier described above can be implemented as a (probabilistic) neural network structure. This neural network has 36 neurons (36 features) in the input layer and 2 neurons (legible and illegible) in the summation layer. The input vector (input layer) is simultaneously distributed to all neurones in the pattern layer. Each neurone in the pattern layer computes a distance between the input vector $\underline{x} = (x_1, \ldots, x_{36})$ and training example $p$ in class $j$. The activation level of this distance measurement is then output into the summation layer. Note that there are only 2 neurons in the summation layer, representing the 2 classes. The summation layers simply sum the inputs from the pattern layers neurones corresponding to the class for which it is trying to compute the probability of a word belonged to specific classes.

## 6. Experiment

Previous work [10] had indicated the need for a careful choice of sample words to allow a good representation of a much larger vocabulary without becoming hopeless unwieldy. Kassel [11] has discussed the design aspects of such data sets and the sample words used in this research are chosen based on that work. The style classification technique was therefore applied on our existing data set, which consists of scanned images obtained from nine writers each containing 150 words at 200x100-dpi resolution.

Tables 2 and 3 show the experimental results obtained from all 36 extracted and the 16 selected features. The first and second columns show the samples that were used as the training data set whilst the third column shows the samples that were used as a test set. The fourth column shows the correct classification results obtained using a common $\sigma$ within the weight function $W$.

Initially the system was trained on the L and IL files containing all 1027 legible (L) and illegible (IL) word images. The classification system was then tested with the same data sets and the results are shown in the first two rows of Table 2 and Table 3. In the second experiment a training data set of 440 word images was randomly selected from the 1647 word images. These were used to derive the Parzen model and the rest were set aside as a test set. These 4 sets are called nL1&nIL1 and nL2&nIL2 respectively were L represents the legible words, IL represents the illegible words and n indicates the number of features.

| Training Set 1 | Training Set 2 | Test Set | % Correct Classification result (common $\sigma$) |
|---|---|---|---|
| 36L | 36IL | 36L | 99% |
| 36L | 36IL | 36IL | 100% |
| 36L1 | 36IL1 | 36L2 | 96% |
| 36L1 | 36IL1 | 36IL2 | 77% |

**Table 2.** Classification result using all 36 extracted features to discriminant between legible and illegible handwriting

| Training Set 1 | Training Set 2 | Test Set | % Correct Classification result (common $\sigma$) |
|---|---|---|---|
| 16L | 16IL | 16L | 99% |
| 16L | 16IL | 16IL | 100% |
| 16L1 | 16IL1 | L162 | 92% |
| 16L1 | 16IL1 | IL162 | 77.5% |

**Table 3.** Classification result using the 16 extracted features to discriminant between legible and illegible handwriting

## 7. Discussion

It can be seen from tables 2 and 3 that the classification performance is 99% and 100% when the test sets are the same as the training set using either the 36 extracted features or the 16 selected features. These tables also show that the classification performance is 96%, 77% and 92%, 77.5% with unseen data using the 36 extracted and the 16 selected features. Experimental results show that all 36 features are needed to get the best classification result.

The results have been analysed to identify the reasons for the misclassification of words. In general most of the misclassified legible words were short (e.g. a, to, etc) and most of the misclassified illegible words were long words (e.g. theoretically, geography, etc).

## 8. Conclusion and future works

This paper has introduced a novel handwriting legibility classification system that can be used to predict the recognition performance of a recogniser for a given handwriting style. Experimental results show that using our definition of legibility of handwriting the best classification result is 86.5% (correct classifications on unseen data), which was achieved by the system using 36 features. Further work will consist of improving the classification using more classes such as middle-legible (i.e. between legible and illegible words). Furthermore the idea of introducing rejection categories will be considered with the view to providing a confidence measure for legibility classification. As mentioned in section 2 the 45 and 75 thresholds were arbitrary choices to provide a starting point. Further work will be done to refine the means of determining these thresholds.

## 9. References

[1] M. Hamanaka and K. Yamada. " On-line Character recognition Adaptively Controlled by Handwriting Quality", seventh international workshop on frontiers in handwriting recognition proceedings, (IWFR' 2000), 33-42, Netherlands, September 2000.

[2] N. Sherkat, T. J. Allen. " Whole Word Recognition in Facsimile Images", Fifth international conference on document analysis and recognition (ICDAR'99), 547-550, India, September 20-22, 1999.

[3] C H Chien and J K Aggarwal.Model. "Construction and Shape Recognition From Occluding Contours". IEEE Transactions on Pattern Analysis and Machine, Intelligence, 11(4): 372-389, 1998.

[4] M. Ebadian Dehkordi, N. Sherkat and R. J Whitrow. "A Principal Component Approach to Classification of Hand-written Words", Fifth International Conference on Document Analysis and Recognition (ICDAR' 99), 781-784, India, September 1999.

[5] M. Ebadian Dehkordi, N. Sherkat and T. Allen, "Case Classification of Off-line Handwritten Words Prior To Recognition", forth IAPR international workshop on Document Analysis system (DAS'2000), 325-334, Rio de Janeiro, December 2000.

[6] H Freeman, "On the Encoding of Arbitrary Geometric Configuration". *IRE* Transactions on Electronic Computers EC-10 (2): 260-268, 1961.

[7] Ferdinand Van Der Heijden, "Image Based Measurement Systems Object Recognition and Parameter Estimation", Wiley, 1995.

[8] B. D. Ripley, "Pattern Recognition and Neural networks", Cambridge, 1997.

[9] R.C. Gonzalez and R.E. Wood, "Digital Image Processing". Addison Wesley, 1993.

[10] M. Jedrzejewski, Automatic Characterization of Handwrting Style, Mphil Thesis, July 1996.

[11] R. H. Kassel, "A comparison of approaches to on-line handwritten character recognition", Doctoral Dissertation, Department of Electronical Engineering and Computer Science, MIT June 1995.

[12] Parzen, E. (1962). " On Estimation of a Probability Density Function and Mode", Annals of Mathematical statistics, 33: 1065-1076.