Off-line Signature Verification using Novel Feature Extraction Techniques and Trajectory Recovery

Vu Minh Nguyen
Master of Information Technology

School of Information and Communication Technology
Science, Environment, Engineering and Technology (SEET) Group
Griffith University

Submitted in fulfilment of the requirements of the degree of
Doctor of Philosophy

May, 2012
Abstract

Feature extraction is an important process in automatic off-line signature verification systems. In this process, only the information that helps to identify the authenticity of questioned signatures is extracted and retained. Amongst the numerous feature extraction techniques investigated by researchers, grid segmentation schemes have been employed more favourably due to their encouraging results. The research presented in this dissertation focuses on improving the performance of Support Vector Machines (SVMs) based off-line signature verification systems using novel feature extraction techniques.

The research began with an in-depth investigation and comparative performance analysis of the Modified Direction Feature (MDF), a structural feature extraction technique. Since the MDF is known for its relatively high accuracies in the cursive character recognition problem, it has been suggested that the performance of the MDF would be as encouraging in verifying Western (English) signatures due to the “cursive appearance” of the signatures. The other features employed for comparative studies were the two grid-based features proposed by Francesco Camastra and Wakabayashi et al. (Gradient feature). The former feature captures the information about the distribution of the foreground pixels in each grid cell whilst the latter utilises the directional information available. The comparisons of these state-of-the-art techniques set the foundation for the development of the novel feature extraction techniques proposed.

In total, three novel local features and four global features were proposed and investigated. The local features include Gaussian Grid, Curvature Map, Variance and the local features are New Ratio, Energy, Trajectory Length, Moment. Amongst the local features proposed, the Gaussian Grid feature significantly outperformed all the state-of-the-art features mentioned above. Nevertheless, the combination of another particularly small-dimensional local feature, the Variance feature, with the global features also outperformed the MDF and the Camastra features, and closely approximates the performance of the Gradient feature. The total dimension of this feature set was only 33 compared to 120 of the MDF. This finding emphasizes the capability of small-dimensional global features.

Apart from the newly proposed feature extraction techniques, this dissertation also presents an investigation of a newly-proposed intersection analysis framework within the context of handwriting trajectory recovery. The proposed framework introduced a new pathway, which may help to tackle the problem of trajectory recovery. This framework was employed to estimate the continuity of contour segments at stroke intersections in the signatures. Experimental results indicated that the curvature information extracted from the undirected estimated contour had improved the performance of the Curvature Map feature, one of the proposed local features. The Curvature Map feature further improved the performance of the Gaussian Grid feature by 0.4% with the average error rate (AER) of 13.54%.
All signature verification experiments of this research were conducted using a subset of the publicly available GPDS-960 signature corpus. With 3840 genuine signatures and 4800 simulated forgeries organized into 160 distinct signatures sets, this database enables more reliable results compared to other publicly available signature database of small size. The public availability of the GPDS-160 signature corpus also facilitates a comparison with the results of other researchers in the field.
Statement of Originality

This work has not previously been submitted for a degree or diploma in any university. To the best of my knowledge and belief, the thesis contains no material previously published or written by another person except where due acknowledgement is made in the thesis itself.

Vũ Minh NGUYEN
Acknowledgements

I am thankful to all the people who have supported me during the course of my candidature. Please understand it is not easy to express my gratitude fully in words.

Above all, I would like to thank my supervisors, A. Prof. Dr. Michael Blumenstein and A. Prof. Dr. John Thornton, for their expertise, guidance, and enthusiasm. My candidature would not be possible without their endless encouragement and support.

I would like to give my special thanks to Dr. Miguel A. Ferrer and the GPDS group for making their high quality signature corpus available to the research community. The results reported in this research would be much less reliable if the experiments were not conducted using the GPDS database.

To Dr. Jolon Faichney, Dr. Vallipuram Muthukumarasamy, Prof. Dr. Graham Leedham, Prof. Dr. Umapada Pal: working with you has always been fruitful and delightful.

To Dr. Adrian Meedeniya, Gervase Tuxworth, and Hemnaphan Suwanwiwat: thank you all for your constructive ideas you shared every time we met.

Special thanks to the administrators of the Griffith High Performance Computing (HPC) - V20z, especially Indy and Paul, who had moved swiftly whenever issues arise. Undoubtedly, the journey would be much more difficult without your assistance.

Last but not least, I would like to give my special thanks to my parents, my brother, my wife, and my beloved son for their endless sources of inspiration, understanding, and support.
# Contents

Abstract i

List of Figures x

List of Tables xiii

Nomenclature xv

List of Publications xvi

1 INTRODUCTION 1
   1.1 Historical Background ............................................. 1
   1.2 Modes of Operation: On-line, Off-line, and Hybrid .................. 2
   1.3 Motivations ...................................................... 3
   1.4 Research Questions and Objectives .................................... 4
      1.4.1 The Modified Direction Feature .......................... 4
      1.4.2 Trajectory Recovery .................................... 5
   1.5 Original Contribution ............................................. 5
   1.6 The organization of the remainder of this thesis .................... 6

2 LITERATURE REVIEW 7
   2.1 Handwriting Identification and Verification ....................... 7
   2.2 Handwriting Models ................................................ 8
   2.3 Types of Signatures ................................................. 9
      2.3.1 Genuine Signatures ...................................... 9
      2.3.2 Forgery .................................................. 10
      2.3.3 Disguised Signature ................................... 11
   2.4 Database .......................................................... 11
      2.4.1 Collection Protocol .................................... 13
   2.5 Preprocessing ...................................................... 13
   2.6 Features Extraction ............................................... 17
      2.6.1 Grid Based and Tree Based Features ..................... 19
      2.6.2 Rotation Invariant Features ............................ 20
      2.6.3 Other Features .......................................... 23
      2.6.4 Feature Vector Dimension Reduction .................... 23
      2.6.5 The Combination and Selection of Features ................ 23
   2.7 Trajectory Recovery ............................................... 24
CONTENTS

2.7.1 Material ................................................. 26
2.7.1.1 Skeleton ........................................ 26
2.7.1.2 Contour ........................................ 27
2.7.2 Preprocessing ....................................... 27
2.7.3 Local Examination ................................. 27
2.7.3.1 Ambiguous Zone Detection ................... 27
2.7.3.2 Ambiguous Zone Analysis ................. 28
2.7.3.3 Double Traced Writing and Hidden Loop Analysis ...... 29
2.7.4 End Points Analysis ............................... 29
2.7.5 Global Reconstruction ......................... 30
2.7.6 Performance Evaluation ......................... 31
2.8 Classification Techniques ......................... 31
2.8.1 Distance Classifiers .............................. 31
2.8.2 Fuzzy Models ..................................... 32
2.8.3 Machine Learning Approaches .................. 33
2.8.3.1 Neural Networks .............................. 33
2.8.3.2 Hidden Markov Models ....................... 34
2.8.3.3 Support Vector Machines .................... 35
2.8.4 Distribution of Inter-class and Intra-class Distances .......... 38
2.8.5 Combination of Multiple Expert Decisions ............ 39
2.9 Performance Evaluation ............................. 40

3 RESEARCH METHODOLOGY AND PROPOSED TECHNIQUES 42
3.1 Research Methodology ............................... 42
3.1.1 Proposed System ................................. 43
3.1.2 Off-line Signature Database .................... 43
3.1.3 Classification Method .......................... 44
3.1.4 Experimental Settings .......................... 45
3.1.5 Evaluation of Verification Accuracy ............. 46
3.2 Intersection Analysis ............................... 46
3.2.1 Boundary Extraction .............................. 48
3.2.2 Determining Contour Curvature Extrema and Isolating Contour Segments .... 49
3.2.3 Candidate Pairs of Segments .................... 51
3.2.4 Bridging Score .................................... 51
3.2.4.1 Line Fitting .................................. 53
3.2.4.2 Circle Fitting .................................. 53
3.2.5 Trajectory Estimation ............................ 53
3.3 Feature Extraction ................................. 54
3.3.1 Local Features .................................... 55
3.3.1.1 Modified Direction Feature .................. 55
3.3.1.2 Camastra Feature ............................ 57
3.3.1.3 Gradient Feature ............................. 58
3.3.1.4 Variance Feature ............................. 59
3.3.1.5 Gaussian Grid Feature ....................... 60
3.3.1.6 Curvature Map Feature ....................... 62
CONTENTS

3.3.2 Global Features ........................................ 63
  3.3.2.1 Ratio Feature ......................................... 63
  3.3.2.2 Energy Feature ....................................... 64
  3.3.2.3 Trajectory Length Feature .......................... 64
  3.3.2.4 Moment Feature ..................................... 67

4 EXPERIMENTAL RESULTS .................................... 68
  4.1 Results of the Proposed Trajectory Recovery Framework .............. 68
    4.1.1 Results of the Curvature Maxima Locator .................... 68
    4.1.2 Bridging Scores Computation ................................ 69
    4.1.3 Trajectory Estimation .................................. 69
  4.2 Results of Local Features ................................ 69
    4.2.1 Results Obtained using the Modified Direction Feature and its variants .... 69
      4.2.1.1 Results Obtained using the MDF for the 75dpi GPDS-160 Corpus 74
      4.2.1.2 Results Obtained Using the e-MDF for the 75dpi GPDS-160 Corpus 75
      4.2.1.3 Results Obtained using the Modified Direction Feature on the 300dpi GPDS-160 Corpus ........................................ 76
    4.2.2 Results Obtained Using the Camstra Feature ................ 78
    4.2.3 Results Obtained Using the Gradient Feature ................ 80
    4.2.4 Results Obtained Using the Variance Feature ................ 80
    4.2.5 Results Obtained Using the Gaussian Grid Feature ........... 81
    4.2.6 Results Obtained Using the Curvature Map Feature ........... 82
  4.3 Results employing Global Features ................................ 84
    4.3.1 Results Obtained Using the Ratio Features .................. 84
      4.3.1.1 Experimental Results of the Ratio Feature Obtained using the 75dpi GPDS-160 Corpus ................. 85
      4.3.1.2 Experimental Results of the Ratio Feature Obtained using the 300dpi GPDS-160 Corpus .................. 88
    4.3.2 Results Obtained Using the Energy Feature .................. 88
      4.3.2.1 Results Obtained using the Energy Feature with the 75dpi GPDS-160 Database ................................. 89
      4.3.2.2 Results Obtained using the Energy Feature with the 300dpi GPDS-160 Database ................................. 90
      4.3.2.3 Results for Off-line Cursive Character Recognition Using the Energy Feature ......................... 90
    4.3.3 Results Obtained Using the Trajectory Length Feature ........ 94
    4.3.4 Results Obtained Using the Moment-based Features ........... 94
      4.3.4.1 Results Obtained Using the Moment-based Feature with One Feature Value ....................... 94
      4.3.4.2 Results Obtained Using the Moment-based Feature with Two Feature Values ....................... 96
  4.4 Results of Feature Fusion .................................. 96
5 ANALYSIS AND COMPARISON OF RESULTS 100
  5.1 Analysis of Trajectory Recovery Results ........................................ 100
    5.1.1 Analysis of Results of the Curvature Maxima Locator Algorithm ........ 100
    5.1.2 Analysis of Results of the Proposed Trajectory Recovery Framework .... 101
  5.2 Analysis of Results Obtained Using Local Features ............................ 101
    5.2.1 Analysis of Results Obtained Using the e-MDF Feature .................... 101
    5.2.2 Analysis of Results Obtained Using the Camastra Feature ............... 104
    5.2.3 Analysis of Results Obtained Using the Gradient Feature ................ 104
    5.2.4 Analysis of Results Obtained Using the Variance Feature ................ 105
    5.2.5 Analysis of Results Obtained Using the Gaussian Grid Feature .......... 105
    5.2.6 Analysis of Results Obtained Using the Curvature Map Feature .......... 107
  5.3 Analysis of Results Obtained Using Global Features .......................... 110
    5.3.1 Analysis of Results Obtained Using the Ratio Feature ..................... 110
    5.3.2 Analysis of Results Obtained Using the Energy Feature ................... 110
    5.3.3 Analysis of Results Obtained Using the Trajectory Length Feature ....... 112
    5.3.4 Analysis of Results Obtained Using the Moment Feature ................... 112
  5.4 Analysis of Results of Feature Fusion ........................................... 113
  5.5 Comparison to Results of Other Researchers ..................................... 113

6 CONCLUSIONS AND FUTURE RESEARCH 115
  6.1 Future Research ................................................................. 116

Bibliography 118
# List of Figures

2.3.1 Scattergraph for the width and height of images from a signature set . . . . . . . . 9
2.5.1 Signature preprocessing ................................................. 14
2.6.1 Ring topology HMM for rotation invariance .......................... 21
2.7.1 Main Components of a Trajectory Recovery System .................. 26
2.7.2 Both start and end points of the word *octopus* are occluded. ... 29
2.8.1 A typical 2-layer MLP .................................................. 33
2.8.2 The flow chart of a typical discrete HMMs signature verification system .... 34
2.8.3 Feature mapping ....................................................... 35
2.8.4 Optimal separation hyperplane ...................................... 36
2.8.5 Mapping from a 2D space to a 3D space ............................ 36
2.8.6 Approach employed by Santos *et al.* ................................ 40
2.9.1 The distribution of score for genuine and targeted forgeries of varied skill . . 41
3.1.1 Proposed Automatic Offline Handwritten Signature Verification System .... 43
3.2.1 A signature image and its boundary representation .................. 49
3.2.2 Example of unbridgable segments. .................................. 51
3.3.1 The extraction of LT and DT values for rows 3 to 5 from left to right .......... 57
3.3.2 A signature and its directional histogram ............................ 59
3.3.3 Colour map representation of the chain code histogram ............... 61
3.3.4 The normalized curvature map of a signature ........................ 62
3.3.5 Ratio Feature .......................................................... 63
3.3.6 Line segments of unequal length represented by the same amount of pixels as an effect of digitisation .......................................................... 65
4.1.1 Missing POIs - 01 to 12 ................................................. 70
4.1.2 Missing POIs - 13 to 24 ............................................... 71
4.1.3 Missing POIs - 25 to 36 ................................................. 72
4.1.4 Missing POIs - 37 to 51 ............................................... 73
4.1.5 Fitting circles at intersections of a signature ........................ 74
4.2.1 The performance of the original MDF in conjunction with other global features . . 75
4.2.2 The performance of the e-MDF feature with varying number of groups using the 75dpi and 300dpi databases .................................................. 78
4.2.3 The FRR and FAR$^2$ curves of the Camastra Feature with various dimensions . . 79
4.2.4 The AERs of the Variance Feature with Different Number of Segments ........ 81
LIST OF FIGURES

4.3.1 The ROC curves of the original MDF in conjunction with R and R2 .......................... 85
4.3.2 The FAR1 rates of the {MDF, R} and {MDF, R2} feature sets as a function of the FRR ................................................................................................................. 86
4.3.3 Performance of Ratio and Ratio 2 with different feature combinations .................... 87
4.3.4 The FAR1 rates of the {Variance6+6, R} and {Variance6+6, R2} feature set as a function of the FRR ................................................................................................................. 89
4.3.5 Performance of energy-based feature in off-line cursive handwritten characters recognition using RBF-NN .............................................................. 92
4.3.6 Performance of energy-based feature in off-line cursive handwritten characters recognition using RBP-NN .............................................................. 92

5.2.1 The performance of the MDF-based feature sets on the 75dpi database .................. 101
5.2.2 The effect of image resolution on e-MDF-based feature sets .................................... 102
5.2.3 Comparative Results of the MDF_L12 and MDF for the 75dpi Database .................. 102
5.2.4 The Performance of MDF Variants on the the 300dpi Database .............................. 102
5.2.5 FAR, FRR^2, and AER of the MDF with different numbers of groups .................. 103
5.2.6 The ROC curves of the MDF and gradient Feature obtained using SVMs and the squared Mahalanobis distance ................................................................. 104
5.2.7 Percentage of total energy retained when a 5 \times 5 Gaussian filter (\sigma = 0.9) is applied in the Gradient feature .................................................................................................................. 106
5.2.8 The performance of the newly proposed feature extraction technique with and without the Gaussian filter ................................................................. 107
5.2.9 ROC curves of the Curvature Map feature for boundary and recovered trajectory input ................................................................................................................................. 109
5.3.1 Performance of energy-based feature for various feature combinations on the 75dpi database .................................................................................................................. 111
5.3.2 Performance of energy-based feature for various feature combinations for the 300dpi database .................................................................................................................. 111
5.3.3 The performance of the Trajectory Length feature in conjunction with other local features for the 300dpi database ................................................................. 112
5.3.4 The performance of the Moment feature in conjunction with other local features for the 300dpi database .................................................................................................................. 112
List of Tables

2.1 Public databases .................................................. 12

3.1 Experimental Settings ............................................. 45

3.2 Experimental Results of Trajectory Recovery Techniques .............. 47

4.1 Results of the Curvature Maxima Locator Obtained Using the US-SIGBASE dataset 69

4.2 Experimental Results of the MDF Feature for the 75dpi Database .......... 74

4.3 Experimental Results of the MDF with 2 Transitions (MDF_L12) ......... 74

4.4 Experimental Results of the e-MDF Feature .......................... 75

4.5 Experimental Results of the MDF Feature using the 300dpi GPDS-160 Corpus ... 76

4.6 Experimental Results of the MDF Feature with the 1st and 2nd layers of transition using the 300dpi GPDS-160 Corpus ......................... 76

4.7 Experimental Results of the MDF Feature with the 2nd and 3rd layers of transition using the 300dpi GPDS-160 Corpus ......................... 76

4.8 Experimental Results of the MDF Feature with the 1st and 3rd layers of transition using the 300dpi GPDS-160 Corpus ......................... 77

4.9 Experimental Results of the MDF Feature with only the 1st layer of transition using the 300dpi GPDS-160 Corpus ......................... 77

4.10 The best AERs of MDF with different layer settings obtained using the 300dpi database ................................................... 77

4.11 Experimental Results of a variant of the MDF Feature with 15 groups and 5 transitions 78

4.12 Recognition accuracies of various feature sets for the C-Cube Corpus 78

4.13 Experimental Results using variants of the Camastra Feature .............. 79

4.14 Experimental Results for the Gradient Feature Obtained Using Squared Mahalanobis Distance classifiers ......................... 80

4.15 Experimental Results for the Gradient Feature Obtained Using SVMs ......... 80

4.16 Experimental Results for the Variance Feature .......................... 80

4.17 Experimental Results Obtained Using the 9 × 9 Gaussian Grid Feature .......... 81

4.18 Experimental Results Obtained Using the 12 × 12 Gaussian Grid Feature .......... 81

4.19 Experimental Results Obtained Using the 15 × 15 Gaussian Grid Feature .......... 82

4.20 Experimental Results Obtained using the Gaussian Grid Feature with varying grid size ...................................................... 82

4.21 Experimental Results Obtained Using the 9 × 9 Directional Map .......... 82

4.22 Experimental Results Obtained Using the 12 × 12 Directional Map .......... 83
4.23 Experimental Results Obtained using a Variant of the $12 \times 12$ Gaussian Grid Feature without Additional Local Features ........................................ 83
4.24 Experimental Results Obtained Using the $12 \times 12$ Curvature Map Feature ........................................ 83
4.25 Experimental Results Obtained Using the combination of the Gaussian Grid Feature and the Curvature Map Feature ........................................ 83
4.26 Experimental Results Obtained Using the $12 \times 12$ Curvature Map Feature Extracted from the Original Contours ........................................ 84
4.27 Experimental Results of the $\{\text{MDF, R2}\}$ feature set for 75dpi GPDS-160 corpus ........................................ 85
4.28 Experimental Results of the $\{\text{MDF, R}\}$ feature set for 75dpi GPDS-160 corpus ........................................ 85
4.29 Experimental Results of the $\{\text{MDF, R, E5}\}$ feature set ........................................ 86
4.30 Experimental Results of the $\{\text{MDF, R2, E5}\}$ feature set ........................................ 86
4.31 Experimental Results of the $\{\text{MDF}_L12, R\}$ feature set ........................................ 86
4.33 Experimental Results of the $\{\text{MDF, R}\}$ feature set with the 300dpi database ........................................ 87
4.34 Experimental Results of the $\{\text{MDF, R2}\}$ feature set with the 300dpi database ........................................ 87
4.32 Experimental Results of the $\{\text{MDF}_L12, R2\}$ feature set ........................................ 87
4.35 Experimental Results of the $\{\text{Variance6+6, R}\}$ feature set with the 300dpi database ........................................ 88
4.36 Experimental Results of the $\{\text{Variance6+6, R2}\}$ feature set with the 300dpi GPDS-160 corpus ........................................ 88
4.37 Experimental Results of the $\{\text{Gaussian Grid, R2}\}$ feature set using the 300dpi GPDS-160 corpus ........................................ 88
4.38 Experimental Results of the Energy Feature using the 75dpi Database ........................................ 89
4.39 Experimental Results of the $\{\text{MDF, E5}\}$ Feature Set using the 75dpi Database ........................................ 89
4.42 Experimental Results of the $\{\text{e-MDF, E5}\}$ feature set ........................................ 90
4.40 Experimental Results of the $\{\text{MDF}_L12, E5, R2\}$ feature set ........................................ 90
4.41 Experimental Results of the $\{\text{E5}\}$ feature set ........................................ 90
4.43 Experimental Results of the $\{\text{Camastra8x8, E5}\}$ feature set ........................................ 91
4.44 Experimental Results of the $\{\text{Variance6+6, E5}\}$ feature set ........................................ 91
4.45 Experimental Results of the $\{\text{Variance6+6, R2, E5}\}$ feature set ........................................ 91
4.46 Experimental Results of the $\{\text{Gradient, E5}\}$ feature set ........................................ 91
4.47 Experimental Results of the $\{\text{GaussianGrid, E5}\}$ feature set ........................................ 91
4.48 Experimental Results of the $\{\text{CurvatureMap, E5}\}$ feature set ........................................ 92
4.49 Experimental Results of the $\{\text{MDF, TrajectoryLength}\}$ Feature Set Using the 300dpi Database ........................................ 93
4.50 Experimental Results of the $\{\text{Camastra8x8, TrajectoryLength}\}$ Feature Set Using the 300dpi Database ........................................ 93
4.51 Experimental Results of the $\{\text{Variance6+6, TrajectoryLength}\}$ Feature Set Using the 300dpi Database ........................................ 93
4.52 Experimental Results of the $\{\text{Gradient, TrajectoryLength}\}$ Feature Set Using the 300dpi Database ........................................ 93
4.53 Experimental Results of the $\{\text{Gaussian Grid, TrajectoryLength}\}$ Feature Set Using the 300dpi Database ........................................ 93
4.54 Experimental Results of the $\{\text{CurvatureMap, TrajectoryLength}\}$ Feature Set Using the 300dpi Database ........................................ 94
4.55 Experimental Results of the TrajectoryLength in conjunction with Other Local Feature using the 300dpi Database ........................................ 94
<table>
<thead>
<tr>
<th>Table Number</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>4.56</td>
<td>Experimental results of the {MDF, Moment1} feature set using the 300dpi database</td>
<td>95</td>
</tr>
<tr>
<td>4.57</td>
<td>Experimental results of the {Camastra8x8, Moment1} feature set using the 300dpi database</td>
<td>95</td>
</tr>
<tr>
<td>4.58</td>
<td>Experimental Results of the {Variance6+6, Moment1} Feature Set Using the 300dpi Database</td>
<td>95</td>
</tr>
<tr>
<td>4.59</td>
<td>Experimental results of the {Gradient, Moment1} Feature Set Using the 300dpi Database</td>
<td>95</td>
</tr>
<tr>
<td>4.60</td>
<td>Experimental Results of the {GaussianGrid, Moment1} Feature Set Using the 300dpi Database</td>
<td>95</td>
</tr>
<tr>
<td>4.61</td>
<td>Experimental Results of the {CurvatureMap, Moment1} Feature Set Using the 300dpi Database</td>
<td>96</td>
</tr>
<tr>
<td>4.62</td>
<td>Experimental results of the {e-MDF, Moment3} feature set using the 300dpi database</td>
<td>96</td>
</tr>
<tr>
<td>4.63</td>
<td>Experimental results of the {Camastra8x8, Moment3} feature set using the 300dpi database</td>
<td>97</td>
</tr>
<tr>
<td>4.64</td>
<td>Experimental Results of the {Variance6+6, Moment3} Feature Set Using the 300dpi Database</td>
<td>97</td>
</tr>
<tr>
<td>4.65</td>
<td>Experimental results of the {Gradient, Moment3} Feature Set Using the 300dpi Database</td>
<td>97</td>
</tr>
<tr>
<td>4.66</td>
<td>Experimental Results of the {GaussianGrid, Moment3} Feature Set Using the 300dpi Database</td>
<td>97</td>
</tr>
<tr>
<td>4.67</td>
<td>Experimental Results of the {CurvatureMap, Moment3} Feature Set Using the 300dpi Database</td>
<td>98</td>
</tr>
<tr>
<td>4.68</td>
<td>Experimental Results of the {MDF, E5, R2, Trajectory Length, Moment} Feature Set Using the 300dpi Database</td>
<td>98</td>
</tr>
<tr>
<td>4.69</td>
<td>Experimental Results of the {Variance6+6, E5, R2, Trajectory Length, Moment3} Feature Set Using the 300dpi database</td>
<td>98</td>
</tr>
<tr>
<td>4.70</td>
<td>Experimental Results of the {Camastra4x4, E5, R2, Trajectory Length, Moment3} Feature Set Using the 300dpi database</td>
<td>99</td>
</tr>
<tr>
<td>4.71</td>
<td>Experimental Results of the {GaussianGrid, CurvatureMap} Feature Set using the 300dpi Database</td>
<td>99</td>
</tr>
<tr>
<td>5.1</td>
<td>Experimental Results of the Gradient Feature</td>
<td>104</td>
</tr>
<tr>
<td>5.2</td>
<td>Performance comparison for the Variance6+6 with global features</td>
<td>105</td>
</tr>
<tr>
<td>5.3</td>
<td>Experimental Results Obtained using the Gaussian Grid Feature</td>
<td>105</td>
</tr>
<tr>
<td>5.4</td>
<td>Verification accuracies obtained using the recovered trajectories and the boundaries</td>
<td>108</td>
</tr>
<tr>
<td>5.5</td>
<td>Statistics of results of the supplementary experiments for the Curvature Map feature</td>
<td>108</td>
</tr>
</tbody>
</table>
Nomenclature

AER  Average Error Rate
AUC  Area Under Curve
COG  Centre of Gravity
DRT  Discrete Radon Transformation
DT   Direction of Transition
DTW  Dynamic Time Warping
EER  Equal Error Rate
FAR1 False Acceptance Rate for random forgeries
FAR2 False Acceptance Rate for simulated forgeries
FAR  False Acceptance Rate
FFT  Fast Fourier Transformation
FNR  False Negative Rate
FPR  False Positive Rate
FRR  False Rejection Rate
FTE  Fail to enrol
HMM  Hidden Markov Model
LT   Location of Transition
MDF  Modified Direction Feature
MLP  Multilayer perceptron
NN   Neural Network
PCA  Principle Component Analysis
POI  Point of Interest
RBF  Radial Basis Function
<table>
<thead>
<tr>
<th>Abbreviation</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>RBP</td>
<td>Resilient Back-propagation</td>
</tr>
<tr>
<td>REF</td>
<td>Ring External Feature</td>
</tr>
<tr>
<td>RIF</td>
<td>Ring Internal Feature</td>
</tr>
<tr>
<td>SVM</td>
<td>Support Vector Machine</td>
</tr>
<tr>
<td>VQ</td>
<td>Vector Quantization</td>
</tr>
</tbody>
</table>
List of Publications


Chapter 1

INTRODUCTION

The human hand has a sophisticated structure consisting of 27 bones and some 40 muscles. The movements of the hand are generated by a series of simultaneous contractions and the relaxation of the muscles around the fingers, hand, wrist, and occasionally the arm. The whole system is coordinated by the central nervous system which is affected by psychological and mental state as well as physical and practical conditions [86]. Broadly speaking, it is believed that the handwritten signatures, being the products of such complex systems, contain personal traits which are unique among individuals. Altogether, the relative stability and difficulty in imitating genuine signatures gave rise to the use of signatures as a biometric.

Compared to other biometrics, signature verification has several advantages including acceptability, collection, and circumvention [92]. Thanks to its convenient nature, automatic handwritten signature verification can have a wide range of applications. Financial institutes can verify bank cheques or credit card transactions automatically. Security systems can use signatures to verify individuals. Organizations can employ automatic signature verification systems to verify certificates and contracts. The construction of an automatic signature verification system based on signature images turns out to be a challenging problem.

This chapter is organized as follows: Firstly, a brief overview on the historical aspects of signature verification is presented in Section 1.1. As this research focuses on off-line signature verification, it is necessary to distinguish the off-line from other modes of operation. This is presented in Section 1.2. After that, Section 1.3 details the motivations of this work. The research questions and objectives follow in Section 1.4. The major original contributions of this work to the body of knowledge is then summarized in 1.5. Finally, the organization of the rest of this thesis is described in Section 1.6.

1.1 Historical Background

Along with the proliferation of handwriting, signatures have been used and widely accepted by society. Signatures are used as a seal of approval and authenticity in government, legal, and commercially important transactions. As a consequence, people with illicit purposes have attempted to forge handwriting as well as signatures. History has taught that the practice of forging documents and handwriting appears as early as the development of writing [86]. The Roman Empire is one of the earliest governments that provided laws for testimony respecting dispute documents by experts. Irrespective of this, experts’ testimony was only first admitted in English-speaking court
two centuries ago, in 1792. This practice only became consistent decades later with the passing of the Common Law Procedure Act in England in 1854.

Verifying handwriting manually has several drawbacks. The accuracy of the judgement is subject to the expertise, mental and physical conditions of the examiner. The verification process, which requires a considerable number of measurements to be carefully performed, is time-consuming. Nevertheless, this professional activity is usually associated with high costs. With the advent of the computer in the 20th century, researchers started to explore the application of computers to automate the task of verifying handwriting to overcome these limitations. Automated signature identification/verification is a research field that attempts to create reliable machines, which can identify or verify human signatures.

Mauceri [140] reported the first work in on-line signature identification as early as 1965. This research employed 2350 genuine signatures produced by 45 writers. The accuracy was reported to be as high as 90%. In 1966, Kozinets et al. [174] started employing electronic computers for an off-line writer authentication system. As the electronic computer became more and more popular, especially with the introduction of personal computers in 1981, automatic writer identification and verification using static signature images became more active [124]. In 1986, Ammar et al. [8] first proposed a pseudo-dynamic feature extraction technique to extract pressure information from grey images. These authors also reported an average error rate (AER) of 5%. In 1987, Sabourin and Plamondon [190] reported the first research, in which the proposed system was tested with 17 professionally produced skilled forgeries. The skilled forgeries were rejected nearly by chance. In 1993, the scientific basis of forensic handwriting analysis techniques was questioned in the well-known case Daubert vs. Merrell Dow Pharmaceuticals [128].

Despite encouraging results being reported, the discipline of automatic handwritten signature verification using static images is considered less mature compared to its dynamic counterpart [79]. It is thus necessary to distinguish between three modes of operation for an automatic signature verification system: off-line, on-line, and hybrid.

1.2 Modes of Operation: On-line, Off-line, and Hybrid

The performance of a signature verification system varies significantly based on the different types of information available to the verification system in each mode. Occasionally, the signature verification problem is mistakenly considered a solved problem due to the high accuracies reported from on-line systems.

Whenever the input information can be represented as a temporal function, the verification system is considered on-line verification. Usually, the on-line stream of information is captured on-the-fly when an individual writes using special hardware such as a stylus and tablet, digitizer pen, or touch screens. The data obtained may consist of various types of information depending on the hardware employed. It can be local pressure, acceleration, speed, number of strokes, and order of strokes. The large performance gap between on-line and other modes of operation is largely due to the availability of one or more of these types of information [124]. It is noted that, on-line data can be used to generate static signature images [36, 181]. This mode of verification is suitable wherever the result is required as soon as clients finish their writing such as points of sale or receptions.

The verification process is called off-line when it is undertaken using the static signature image solely. Unlike its on-line counterpart, the off-line mode does not require any specialized hardware.
The trade-off is that the amount of information obtained is two orders greater but much more difficult to be interpreted. Besides, traces of dynamic information are almost absent or very difficult to recover. The recovery of such information requires professional skills and techniques whose implementation on computers is deemed to be challenging [80, 201]. These disadvantages supposedly obstruct off-line systems from producing good recognition results. Some investigations performed by expert document analysts also suggested that the detection of skilled forgeries requires not just static information but also dynamic information [174].

In the hybrid mode, the verification of signature image is performed with reference to the previously registered on-line data. This approach often includes the estimation or recovery of the trajectory from the scanned image prior to comparing the properties of the recovered trajectory against the profile established before. Notable investigations employing this approach include those conducted by Qiao et al. [180] and Zimmer and Ling [231, 233].

1.3 Motivations

With the global growth in security concerns in recent years, it is necessary to develop more accurate, convenient, and economic security systems. The belief that a system meeting these requirements could be constructed by utilising human signatures poses great motivations to the present research. Not only the benefits that the distinct characteristics of signatures could bring about but also the challenging aspects of the problem are of interest.

Unlike other authentication protocols using passwords, access cards, or PIN codes, the ability to sign possessed by a human is unlikely to be lost, forgotten, or stolen [176]. Signatures are unique amongst people and difficult to imitate. A normal individual should be able to produce his/her signatures at any time and anywhere upon request as a proof of identity. Research results also indicate that the reproducibility of signatures does not differ within age-groups in the population [70]. Moreover, off-line signature verification does not require special sample capturing devices as in hand vein, fingerprint, and retina recognition. The popularity of high resolution digital imaging devices such as scanners, digital cameras, web cams, and mobile phones enables the construction of mobile and economic off-line handwritten signature verification systems.

Theoretically speaking the construction of signature verification systems, especially off-line systems, are inexpensive. Imaging-capable gadgets are already popular and can be purchased at affordable prices. The processing power of such devices are already high enough to perform demanding tasks such as 3D-gaming or video conferencing. With the advent of cloud computing, resource demanding operations of a verification system can alternatively be performed remotely if necessary.

Once successfully constructed, automatic off-line signature verification systems can be employed at any security check point where the handwritten signatures are currently verified manually. Potential applications include: the validation of personal cheques, credit card transactions, legal documents, contracts, historical documents, etc. However, the verification of human signatures using static images is not a trivial task.

Human signatures do vary although they look similar. It has been observed that there is significant variability in human signatures depending on country of origin, age, time, habits, psychological or mental state, physical and practical conditions [86]. From the perspective of forensic science, it would take significant training over a course of many years for an apprentice to become a forensic document examiner. As a consequence, verifying signatures is considered a professional
task. Moreover, there are not many feature extraction methods, which are being employed by forensic document examiners, that can be computed conveniently by an automatic signature verification system [165]. This contributes greatly to the challenging nature of the automatic off-line signature verification problem.

From the machine learning perspective, signature verification is distinct from many other two-class classification problems [173]. Firstly, only the specimens of the genuine signature class are collectible. Secondly, the learning process of a practical verification system requires that the number of genuine specimens be small. These restrictions make signature verification even more challenging. The basic axiom of handwriting identification as proposed by McCarthy [141] was that “no two writings by the same or different persons are identical”. Signatures are complex artificial patterns with variations arising from many factors such as mental and physical conditions, instruments used, etc. Such multi-source variations enlarge the feasible set and increase the probability that forgeries may fall into.

The challenging nature of the problem also arises from the fact that forgeries can be of a highly-skilled level. Totty [209], Buglio and Gidon [29] reported cases in which the forgeries overcame the common symptoms of forgeries, such as poor line quality, different overall pictorial shapes and sizes, retouching, etc., and could only be identified by competent professionals after thorough examination. Consequently, an ideal signature verification system should effectively detect skilled forgeries and be tolerant to the variations of genuine signatures simultaneously.

Another obstacle for signature verification research is the unavailability of a standard signature database. The first reason is contributed by the privacy aspect of signatures. Not many people are willing to make their signatures available to the public especially for the purpose of practising forgeries of their signatures. This restriction prevents many databases from becoming available to the research community which, consequently, makes comparative analyses of reported techniques less reliable. The second reason is that it is not easy to recruit professional forgers to produce skilled forged signatures. Although the construction of a readily available signature database allows virtually unrestricted practice for imitation, lay forgers are often lacking in forging experience and motivation. They are not aware of the patterns that questioned document examiners will look at in the process of identifying forgeries.

In the age of information technology, where trends have turned towards automation, the tremendous benefits as well as the challenges discussed above have been a true motivation for researchers.

1.4 Research Questions and Objectives

Although the ultimate objective of research in automatic off-line signature verification is to create a practical verification system which is capable of reliably distinguishing genuine signatures and forgeries, the scope of this research is limited to addressing a number of challenges as part of this thesis.

1.4.1 The Modified Direction Feature

The Modified Direction Feature (MDF) is a feature extraction technique in the area of pattern recognition. It already has achieved promising results in the area of cursive character recognition [23] as well as other pattern recognition problems [68, 67, 66]. Due to the nature of skilled hand-
writing movement found in both cursive handwriting and signatures, it is suggested that the MDF could help to increase accuracies in the off-line signature verification problem. Preliminary results indicate that the Support Vector Machine classifier [96], when operated in conjunction with the MDF and other global/local feature extraction techniques, could obtain comparable accuracies to those of other state-of-the-art feature extraction techniques. Naturally, it is encouraging to have an in-depth investigation into the MDF feature extraction technique and its variations.

The research questions regarding the MDF feature extraction technique are:

1. What modifications can be made to the MDF itself in order to improve its performance in off-line signature verification?

2. Can the performance of an MDF-based off-line signature verification system be improved by employing additional global or local features?

1.4.2 Trajectory Recovery

Apart from interest in performance improvement of the Modified Direction Feature, this research also investigates whether the recovery of on-line information from signatures, specifically the recovered pen trajectory, significantly increases the accuracy of an off-line signature verification system. The main research questions concerning the application of trajectory recovery for signature verification is:

- Can the performance of a signature verification system be improved by employing features extracted from the recovered trajectory?

1.5 Original Contribution

The major contributions are:

1. An equal size segmentation technique that significantly improves the performance of the MDF feature on low resolution images.

2. A novel handwriting intersection analysis framework which includes:

   (a) A novel technique for locating the local curvature maxima points

   (b) A novel scoring technique to determine the possibility that two contour segments are paired

3. A set of novel feature extraction techniques for off-line signature verification including 3 local and 3 global features. Amongst which are as follows:

   (a) The Gaussian Grid feature: a novel grid-based feature extraction technique that outperforms other state-of-the-art local feature extraction techniques such as the MDF, Camastra’s feature, and the Gradient feature.

   (b) The Curvature Map feature: a novel curvature-based feature extraction technique, which utilises the recovered trajectory and further improved the performance of the Gaussian Grid feature.

   (c) A novel set of local and global features that outperforms the Modified Direction Feature and Camastra features in terms of both accuracy and feature vector dimension.
1.6 The organization of the remainder of this thesis

The remainder of this thesis is organized into the following five chapters. Chapter 2 thoroughly reviews the approaches and techniques employed in the area of automatic off-line handwritten signature verification. Chapter 3 details the research methodology and the proposed feature extraction techniques. The comprehensive set of experimental results is presented in Chapter 4 before being analysed in depth and compared to results from other researchers in Chapter 5. Finally, the concluding remarks as well as recommended directions for further investigation are presented in Chapter 6.
Chapter 2

LITERATURE REVIEW

Off-line signature verification has existed on the frontiers of handwriting processing research for several decades. As evident from the literature, researchers in the area have investigated a considerable number of approaches. The state-of-the-art in signature verification up until the year of 2000 was summarized and presented in a series of comprehensive surveys by Plamondon et al. [173, 124, 176]. A large bibliography of more recent works in the area can also be found in the brief survey of Impedovo and Pirlo [87]. In the present chapter, the relevant literature will be systematically reviewed in more detail.

This chapter is organized into the following sections: Firstly, the broader but closely related area of handwriting verification is briefly reviewed in Section 2.1. Section 2.2 briefly reviews the two major handwriting models. The various types of signatures are described in Section 2.3. After that, the issues concerning signature databases are revisited in detail in Section 2.4. Section 2.5 details the preprocessing techniques, which are frequently employed by researchers. Section 2.6 is devoted to feature extraction, the crucial component of a complete signature verification system. Section 2.8 reviews the approaches to the problem. Finally, Section 2.9 reviews various aspects of performance evaluation in signature verification.

2.1 Handwriting Identification and Verification

The consultation of forensic document examiners has long been in practice in courts of law to help determine the authenticity of questioned documents [86]. Researchers such as Kam et al. affirmed that the recruitment of professionals is an appropriate practice [103, 104, 105]. Using a number of property extraction techniques, the examiners gather qualitative and quantitative measurements regarding questioned documents and, consequently, come up with conclusions about the authenticity of the handwriting [78]. In [86], Huber and Headrick summarized a list of twenty-one handwriting discrimination elements used by forensic document examiners. Among those, Leedham and Pervouchine [128] concluded only a few were described with sufficient detail so that they can be implemented by automatic extraction systems.

With the advent of the personal computer, software systems have been developed to assist forensic document examiners. Manual examination of documents is a meticulous task that requires a large number of careful measurements. A well known system for that purpose is CEDAR-FOX developed by Sripri et al. [203]. In the identification mode, a questioned document is examined, in terms of similarity, against a set of writings from known individuals. In verification mode,
CEDAR-FOX provides a quantifiable level of confidence whether the document being examined and a known document were written by the same person. For further examination of the questioned document in analysis mode, this system was enhanced with various image processing utilities.

Despite the developments mentioned above, the scientific validity of the analysis made by document examiners has recently been questioned. Compared to other forensic sciences, such as fingerprint or DNA analysis, many forensic handwriting examination techniques have far less scientific support although they may appear to be persuasive [128]. In their research into the validity of the use of handwriting as a biometric, Leedham and Pervouchine [128, 165] concluded that it is impossible to determine whether handwriting is a biometric or not, given the current understanding of handwriting.

2.2 Handwriting Models

In a series of articles published from 1995 to 2003, Plamondon [169, 168, 170, 171] introduced the Kinematic Theory of Rapid Human Movements. It was proposed that handwriting is controlled by a sequence of vectorial commands. Each of these commands stimulates a neuromuscular system which subsequently produces an impulse. The amplitude of an impulse converges to a log-normal function. A rapid movement is controlled simultaneously by two opposite systems, the agonist and the antagonist. As these two systems do not execute in perfect opposition, the summation of these two systems produces variations in velocity.

In 1995, Guerfali et al. [69] demonstrated that the delta-lognormal model is capable of generating and representing complex handwriting movements in their research. In such a model, the movement of the hand can be regarded as a system of push-pull impulses controlled by the neuromuscular networks. An interactive tool for the generation of handwriting and signatures using the log-normal model was also introduced by Djoua et al. [43] in 2006. More recently, Plamondon et al. [172] demonstrated that the extraction of log-normal model parameters from handwritten strokes using online data is realistic.

The application of the delta-lognormal model for the signature verification problem has recently been investigated by Zimmer and Ling [233]. In their hybrid system, each user produces a number of specimens using a digitizer tablet in the registration phase. The registered online data allows the extraction of information from signature strokes. The position and dimension of the rectangle enclosing major strokes is located and adjusted. From these rectangles, as well as the whole signature, global and local features are extracted. The reference model of a user is then created using the standard deviation of each feature extracted. In order to verify the authenticity of a questioned signature, the same number of features is extracted and compared with the reference model using a Euclidian distance classifier. A verification accuracy of up to 99% for random forgeries, using 10 reference genuine signatures, has been reported.

Apart from the log-normal model, researchers such as Hala Bezine et al. [21, 20] also investigated the simulation of rapid handwriting using the Beta-Elliptic model. This model was based on two principle assumptions, including: (i) Fast handwriting is partially pre-programmed like other highly skilled motor processes (ii) Handwriting movements are planned and represented in the velocity domain. The major difference between the Beta-Elliptic and the log-normal model is that, in the Beta-Elliptic model, the global impulse response converges to a beta curve instead of a log-normal curve with a sufficiently large number of neuromuscular sub-systems. Despite encouraging results with on-line handwriting, the investigations of handwriting models using off-line
images are still very limited. It is quite difficult to recover the dynamic parameters of strokes from handwriting images when the velocity information is unavailable.

2.3 Types of Signatures

Research in signature verification usually involves three main types of signatures: genuine signatures, forgeries, and disguised signatures. Whilst a certain degree of stability is observed in genuine signatures, the forgeries produced by the same forger exhibit significant variations depending on the skill of the forger as well as the amount of information about the genuine signature target that is exposed to the forger. It is widely agreed that the intra-personal variance is smaller than the interpersonal variance [173]. The verification of signatures is possible only if this crucial assumption holds. Figure 2.3.1 shows the distribution of the width and height dimensions of the genuine signatures against simulated forgeries for a particular writer from the GPDS signature corpus.

2.3.1 Genuine Signatures

When an authentic writer produces his own signatures under normal conditions, the signature obtained is called a genuine signature. Being unrestricted by any rules, genuine signatures can be considered free drawings and do not necessarily convey any meaning. In many instances, genuine signatures are unreadable. Although the signatures of an individual may appear very similar, it is widely agreed that signatures are produced differently each time a person signs. In other words, there exist no two signatures which are geometrically identical. Hence, when presented with two identical signatures, at least one of them must be a forgery, i.e. traced or photocopied [86]. The intra-personal stability and variability of genuine signatures are affected by many factors such as country, age, time, habits, psychological or mental state, physical and practical conditions[174].

Genuine signatures can only be produced when the subject is conscious and willing to write in the usual manner. This is different from some other forms of biometric such as fingerprints or DNA. Fingerprints or DNA can still be used for authentication when an individual is unconscious.
2.3.2 Forgery

Forged signatures form the handwriting of impostors with the purpose of being falsely recognized as genuine signatures of another individual. Generally speaking, the dissimilarities between forgeries and genuine signatures mainly originate from the differences between the skilled motor programs responsible for the generation of signatures in the brains of authentic writers and forgers.

Compared to genuine signatures, individual characters or components of the forgeries tend to be larger. The curves may become angles and vice versa. Strokes may be terminated suddenly when they should be smooth. Continuous strokes can be replaced by hesitation and retouching. Redundancy such as strokes or even characters occurs. Quality of lines can be poor; Other differences that may occur include punctuation, local or global pressure, baseline, or spacing. Personal writing characteristics of the imitators can even be exposed in their forgeries. Many researchers believe that the majority of these characteristics cannot be modelled and computed for automatic signature verification [137].

As mentioned previously, it is widely agreed that the above characteristics of forgeries mainly originate from the differences between the skilled motor programs in the brains of authentic writers and forgers. The genuine signatures are produced by consistent, skilled, and smooth automated executions of a chain of motor commands in the brain of the authentic signer. Conversely, the visual feedback mechanism in the forger’s brain interferes with the signing process and causes inconsistent, unskilled, and hesitance sub-commands. However, the forgers can become more skilled through practise. Interestingly, significant improvements were observed if they had been motivated [28].

As summarized in [153], the forgeries belong to one of the following six categories ordered by level of verification difficulty:

1. Forgeries produced without the knowledge of either writer’s name nor the signature image of the targeted individual: They may significantly differ from genuine signatures in both size and shape and are very easy to recognize. The forgery can even be forger’s own genuine signature. In the literature, they are often named Random Forgery [98], Zero Effort Forgery, Simple Forgery [88], or Substitution Forgery [174].

2. Forgeries produced with knowledge about the genuine writer’s name only: Hanmandlu et al. [75] categorized this type of signature as a Random Forgery whilst Justino et al. [98] categorized this type as a Simple Forgery. Occasionally, some researchers may call these a Casual Forgery [79]. This type of forgery is supposed to be the most popular although they are not hard to detect.

3. Forgeries produced by inexperienced forgers without the knowledge of their spelling after having observed the genuine specimens closely for some time: are categorized as Unskilled Forgeries by Hanmandlu et al. [75].

4. Forged signatures produced after examining closely and practising unrestrictedly with the images of genuine signatures by non-professional forgers are categorized as Freehand Forgery [174], Simple Forgery/Simulated Simple Forgery [88, 57], and a Targeted Forgery by Huang and Yan [84].

5. Skilled forgery refers to the forgeries produced by professional forgers or people possessing knowledge in handwriting analysis or experience in copying signatures [75]. Examining this type of forgery is the most challenging task even for professionals as their appearance resemble
genuine signatures and have overall pictorial accuracy. In their signature verification research using forgeries produced by professional forgers, Sabourin and Plamondon [190] reported that the FAR for skilled forgeries was as high as 47%. There is little doubt that the highly skilled forgeries are potentially more stable than genuine signatures. Totty [209] reported the case of an individual whose forgeries exposed no common symptoms of forgery such as tremor, poor line quality, hesitation, or pen lifts.

6. Forgeries produced by tracing a genuine signature: Huber and Headrick [86] called them Traced forgery. Forgeries of this type cannot be detected without detailed examination as their shape, size, and line trajectory are identical to genuine signatures. Consequently, automatic detection of this forgery type requires the questioned signature to be acquired in colour at a higher resolution and must be done at line quality level.

It is suggested that different types of forgeries may require a different verification approach. Whilst the verification accuracy for random forgeries has reached an error rate below 0.1% in the literature, conversely the verification accuracy of targeted and skilled forgeries remains a challenging problem.

2.3.3 Disguised Signature

There are situations whereby the signatures were unwillingly produced, i.e. under another’s pressure, by an authentic signer with an intention to reject the authenticity of the signed document afterwards [86]. This type of signature is named disguised signature. Although disguised signatures are produced by authentic signers and resemble genuine signatures, they contain features which are often found in forgeries. This type of signature has recently been brought to the attention of the automatic signature verification community by the 4NSigComp2010 signature verification competition [133] at the ICFHR ’10 international conference.

2.4 Database

Signature databases plays an important role in signature verification research. However, there has been no standard database that is widely accepted, although research in the area has been actively pursued for many years. The majority of reported works in the area have been conducted using proprietary databases of limited size and quality. As a result, it is difficult to compare the reported techniques and results employing different databases directly. The major factors obstructing the creation of a standard signature database for scientific research are the large quantity and quality requirements.

To be suitable for research, a signature database should have a large number of signatures collected from many people so that it can represent the population. The large number of signatures also helps to produce more statistically reliable experimental results. Since signatures of an individual may vary noticeably over time, ideally, the genuine signatures of contributors should be collected in different sessions. However, the organisation of multiple sessions for each contributor appears to be too costly [213].

From the literature it can be seen that a large portion of researchers employed relatively small proprietary databases. Relatively large signature databases such as the GPDS or MCYT have only only been available for a few years. This is mainly due to the labour intensive nature of the collection work. Table 2.1 presents detailed information about the publicly available signature databases.
In order to correctly classify a questioned pattern, an automatic pattern recognition system must refer to its knowledge base which has been previously established. This knowledge base should be created using easily-collected samples such as genuine signatures, random forgeries, or synthesized signatures. It is impractical to collect targeted forgeries, especially skilled ones on a large scale. Moreover, the number of genuine specimens employed for the establishment of a signature model should also be minimized due to the fact that in real-world situations, these specimens are sparsely available. These two restrictions make signature verification different from other popular two-class classification problems where samples of both classes are often widely available and collectable.

Many researchers attempted to address the problem of insufficient number of genuine samples for training of an automated system. While some converted an on-line database into an off-line database [36], others generated additional training samples, or artificially generated training samples, by distorting or combining genuine signatures. In their research, Fang et al. [54] applied the two-dimensional elastic matching method on pairs of genuine signatures to generate additional signatures. Firstly, the skeletons of a signature pair are extracted. After that, the first skeleton is put over the second one, whereby it is fixed and shifted around in order to minimize the value of a function which calculates the degree of mismatch. Once the two skeletons are best matched, the displacement in position between the pair of corresponding segments in the two signatures is defined as a displacement vector. An intermediate signature is then obtained by deforming one of the genuine sources along the displacement vector. As this signature is not always perfectly smooth and connected, further morphological operations of closing are applied to bridge the gaps and fill the holes in order to complete the process of generating an additional signature.

In another work by Huang and Yan [84, 83], simulated genuine samples are generated directly by slightly perturbing a genuine signature. The proposed perturbations include slant distortion, size distortion horizontally and/or vertically, rotation distortion, and perspective view distortion. However, this approach to the problem of lacking training samples is controversial as there is no way to determine whether a simulated sample is actually closer to the genuine class or forgery class. Finding the answer to this problem is actually as difficult as the signature verification problem itself.

Another aspect concerning the construction or the choice of a signature database is the size that the database should be in order to provide statistically significant results when comparing two different approaches. The organization of signature collection sessions with a number of writers large enough to represent the population over a long period of time, is not a trivial task. The writers should be from both genders, of different age ranges, and from different backgrounds. Guyon et al. [71] noted the “chicken and egg” nature of choosing the appropriate test size for a problem since it

<table>
<thead>
<tr>
<th>Database</th>
<th>Resolution</th>
<th>Sets</th>
<th>Gen.</th>
<th>For.</th>
<th>Employed by</th>
</tr>
</thead>
<tbody>
<tr>
<td>GPDS-39</td>
<td>75dpi</td>
<td>40</td>
<td>24</td>
<td>30</td>
<td>Armand et al. [11, 12]</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Larkins and Mayo [121]</td>
</tr>
<tr>
<td>GPDS-100</td>
<td>600dpi</td>
<td>100</td>
<td>24</td>
<td>24</td>
<td>Vargas et al. [215]</td>
</tr>
<tr>
<td>GPDS-160</td>
<td>300dpi</td>
<td>160</td>
<td>24</td>
<td>30</td>
<td>Martinez et al. [139]</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Ferrer et al. [57]</td>
</tr>
<tr>
<td>GPDS-960</td>
<td></td>
<td>960</td>
<td>24</td>
<td>30</td>
<td>Vargas et al. [213]</td>
</tr>
<tr>
<td>MCYT-75</td>
<td></td>
<td>75</td>
<td>15</td>
<td>15</td>
<td>Alonso-Fernandez et al. [7]</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Wen et al. [221]</td>
</tr>
</tbody>
</table>

Table 2.1: Public databases
is impossible to determine the statistical significance of a database of size \( n \) before obtaining the performance of the two systems being compared. Despite that, the relationship between sample size \( n \), level of significance \( \alpha \), error rate \( p \), and the proportion \( \beta \) of error margin \( \epsilon(n, \alpha) \) and \( p \) has been established in the following equation:

\[
 n = \frac{2\ln(\alpha)}{\beta^2p} \tag{2.4.1}
\]

### 2.4.1 Collection Protocol

The collection protocol has a significant impact on the quality of a signature corpus. For example, the performance of systems employing grey images tends to be less stable in a database created using different types of pens [215]. If an individual is motivated by rewards, he/she is more concentrated and his/her signatures are likely to be more stable. Similarly, if an impostor is motivated, it is likely that his/her skill will improve through practice [28].

In order to create a high quality signature research database, the quality of the signatures collected should also be carefully controlled. Although genuine signatures are relatively easier to collect at a high level of quality, it is agreed that the performance of signature verification systems can be affected significantly by the substantial signature inconsistency coming from a few individuals [174]. One way to handle this situation is to deny the enrollment of users whose signatures are highly inconsistent. The other workaround is to relax the decision threshold so that the system could accept their signatures. The trade-offs of these two approaches are robustness and security.

Inconsistencies may also occur during the enrollment of individuals. The inclusion of the outliers in the training process is prone to negatively affect the representativeness of a writer’s profile. Therefore it is preferable that the outliers are identified and excluded during the establishment of a writer’s profile. Wen et al. [221] proposed a genuine signature quality assessment technique for the selection of genuine specimens for training. Specifically, the researchers suggested that specimens should only be employed for training if their geometric distance to the mean distance divided by the mean distance between genuine specimens does not exceed a predefined threshold. Significant improvements were also reported in their work.

The collection of highly skilled forgeries is relatively harder due to the unavailability of highly skilled forgers for recruitment. As Huber and Headrick [86] commented, there would be no interest for competent people to offer their services. Customers with illicit intentions would not take the risk of being blackmailed permanently once engaging in such services. Consequently, the creation of forgeries in research databases are usually performed by lay forgers (ordinary people). Some researchers [124, 143] suggested the use of simulated forgeries to overcome the scarceness of skilled forgeries. In those works, the genuine signatures were utilized to generate simulated forgeries.

### 2.5 Preprocessing

Normally, signature images are acquired in different formats and resolutions. They need to be processed to enable more accurate feature extraction. They may also contain unexpected marks, stains, or noise which would have negative effects on the recognition accuracy. Preprocessing includes operations for the elimination of such noise and converting the image to a format suitable for the extraction of discriminating information. Han [73] enumerated the commonly adopted steps
in preprocessing: background subtraction, thresholding, noise cleaning, gap filling, and skeleton extraction. Depending on the requirements of subsequent processes, one or more preprocessing algorithms can be omitted and others can be employed. Figure 2.5.1 depicts popular operations in signature image preprocessing.

![Signature Preprocessing Diagram](signature_processing_diagram.png)

Figure 2.5.1: Signature preprocessing

Researchers generally agreed that the patterns should be acquired at an appropriate resolution. On the one hand, low resolution causes information loss. On the other hand, excess resolution is commonly associated with extra storage and can introduce noise, which may negatively affect the feature extraction process. In their research [214], Vargas et al. investigated the impact of acquisition resolutions on the accuracy of a particular off-line signature verification system. The signature images were acquired at resolutions ranging from 45dpi to 600dpi. A polar coordinate-based feature extraction technique was employed in conjunction with the HMM classifiers in that system. Their experimental results indicated that no significant improvements were obtained at resolutions greater than 150dpi despite higher computational cost. These researchers suggested that the appropriate resolution for signature image acquisition is 150dpi.

After the images have been acquired, the signatures need to be extracted from the image background. This is due to the fact that signatures are not always produced on a perfectly white and clean writing surface. Guidelines, watermarks, security patterns, stains, just to name a few, are commonly seen elements of a signature background. Ideally, anything that does not belong to the signature will be excluded from the image area containing the signature prior to further processing. However, background subtraction is not a trivial problem, especially for backgrounds with complex patterns and colours [41, 42, 75]. It has been an active research subject since the dawn of automatic handwriting processing.

The acquired image usually is either a colour image or a grey-level image. The choice of format depends largely on the requirements of the feature extraction method employed subsequently. Some researchers adopted a grey-level format [8, 9]. In a grey-level image, each pixel is assigned a value from 0 to 255 depending on its brightness. A grey-level image can be transformed further into a binary image. Despite disadvantages such as loss and distortion of information or the introduction
CHAPTER 2. LITERATURE REVIEW

of noise, it is agreed that binary images require less storage and their manipulation is less expensive.

According to a survey conducted by Trier and Taxt in 1995 [210], binarisation methods can be
categorized as global or locally adaptive. The former decide a unique threshold which is applied to
all the pixels based on the consideration of the whole image whilst the latter finds a threshold for
each pixel with the consideration of its neighbours. Researchers also compared the performance
of different algorithms using grey-scale images of low contrast with variable background intensity
and noise. The results indicated that locally adaptive techniques performed better than global
techniques in terms of accuracy in exchange for their higher consumption of computational power.
To be more specific, the best locally adaptive algorithm performed 10 times slower than its global
counterpart (3 seconds compared with 0.3 seconds). Among the best locally adaptive algorithms,
there is no single method that outperforms other methods for every image. Moreover, the highest
accuracy obtained still does not meet the requirements of further processing. This observation
partly explains why there are still researchers exploring feature extraction techniques using grey-
level images. From the research of Trier and Taxt, it was found that Otsu’s algorithm [158] was the
best global thresholding algorithm in terms of both speed and accuracy. This algorithm calculates
the global threshold by considering the zeroth and the first-order of cumulative moments of the
grey-level histogram. When taking into account the fact that the handwriting or signatures are
often produced on clean white paper, researchers agreed that Otsu’s algorithm is generally good
enough for the recognition of handwriting [22].

In the third step, irrelevant information introduced during the image acquisition or thresholding
processes such as ‘single’ black pixels on the background or ‘single’ white pixels on the foreground
are removed. The popular method is to search and remove foreground pixels that have less than
two neighbours [14].

The next step in the preprocessing phase is contour smoothing. The signature image obtained
from the thresholding process may contain noise, especially in the buffer zones between the fore-
ground and the background. Such noise should be removed in order to achieve a smooth boundary.
Donggang and Yan [48] suggested a set of rules to smooth the boundary which utilized its Freeman
chain code [60] representation. Such rules look for sets of three or four consecutive points that form
rough segments on the boundary and fix them. Since this algorithm only considers a fixed number
of consecutive points, Hu et al. [81] extended Yu and Yan’s rules with the introduction of the
multipoint smoothing algorithm by treating consecutive primitive segments of the same direction
as a primitive segment. The binarisation process not only introduces noise but may also result
in broken strokes [199]. Consequently, image enhancement and stroke restoration are required.
Shi and Govindaraju [197] proposed an image enhancement algorithm by selective region-growing.
Siyuan and Srihari [199] improved this technique by employing a larger selective region with the
size $7 \times 7$ and reported that foreground pixels separated by further chessboard distance (2) can
be connected. As human beings are capable of recognizing different objects using boundary
information or external boundaries solely, many researchers believe that handwriting recognition can
successfully rely on this information.

After noise of various types is removed and broken strokes are repaired, the contour or external
boundaries are extracted. Some feature extraction techniques only utilise the external boundary
[229] whilst others employ the boundaries of the contours [25]. Unlike in the information that
remains following skeletonisation or thinning, the information pertaining to stroke width is retained
in the contour boundaries. Features extracted from the contours are more sensitive to stroke
width. Chen and Srihari [199] extracted the external boundary of signatures in order to apply
string-matching algorithms, which were often seen in on-line verification techniques. From the binary representation of the signature image, boundary extraction is performed in two steps [162]. Firstly, every inner pixel, specifically the foreground pixel that borders four other foreground pixels, is identified using a raster scan. Secondly, the identified pixels are assigned the same value as background pixels. The remaining black pixels form the boundary.

The apparent drawback of approaches employing the external boundary is the de-emphasis in consideration of internal information. Apart from that, minor variations in signature's strokes or trajectories would significantly affect the signature boundary.

Since a signature may have been produced on a skewed sheet or paper, the acquired image may also need to be skew corrected. This operation is necessary in order to stabilize feature extraction, as suggested by Wessels and Omlin [222]. In their on-line signature verification research, the signature image was rotated about the Centroid by the angle $\theta$, which was calculated mathematically, so that the total horizontal deviation of sample points regarding the Centroid is maximized.

$$\max_{\text{w.r.t. } \theta} \text{var} (x_i^*) \quad (2.5.1)$$

where

$$\begin{bmatrix} x_i^* \\ y_i^* \end{bmatrix} = \begin{bmatrix} x_i \\ y_i \end{bmatrix} \times \begin{bmatrix} \cos \theta & \sin \theta \\ -\sin \theta & \cos \theta \end{bmatrix}$$

The horizontal variance maximization problem is equivalent to the computation of the largest principle vector in Principle Component Analysis (PCA). Subsequently, PCA has been employed by researchers in the area, such as Bansal et al. [15].

Depending on the subsequent feature extraction technique, the signature image can be dilated or thinned. For example, Ueda [211] normalized the width of the strokes in Japanese signatures by blurring the thinned image. Significantly better verification accuracies were reportedly obtained compared to conventional approaches [226]. The signature image can also be thinned successively in a process called skeletonisation. The resulting pattern is called the skeleton and it contains important structural information about the pattern. Moreover, features extracted from a signature skeleton tend to be invariant to writing instruments and materials [14]. At first glance, skeletonisation seems to be easy. In an early work in 1984, Zhang and Suen [228] proposed a fast parallel algorithm for thinning digital patterns. In fact, the outputs obtained were often found not to conform to the source pattern, as noted by Lee and Pan [127]. Some researchers also suggested a set of heuristic rules, while others such as Baltzakis and Papamarkos [14] employed the skeletons of signature images for feature extraction. Their three-step technique included the following tasks: (1) mark all the black pixels, specifically those having one background and two foreground neighbours, (2) examine and convert the marked pixels into background pixels along the contour of the signature image and remove those whose removal would not break the pattern, (3) repeat the process if at least one pixel was changed into a background pixel. Some skeleton extraction techniques can be used in two ways, in the sense that the original pattern can be correctly reconstructed using the skeleton. Beside the skeleton extraction techniques, which facilitate the extraction of stroke width tolerant features, researchers in the area actively search for techniques that effectively recover the trajectories of the writing. In [127], Lee and Pan proposed a technique to trace handwriting trajectories and represent off-line signatures. Firstly, this technique extracts and repairs the skeleton from the handwriting image. A heuristic-rule-based algorithm is then employed to trace the sequences and trajectories from this enhanced skeleton. Finally, the critical points (terminators,
junctons, points whose instantaneous curvature change exceeds certain thresholds), with regards to different curvature thresholds, are extracted. The representation extracted with this technique is invariant with respect to translation, rotation, and scaling. The proposed technique can also properly trace two touched characters without requiring a pre-segmentation process. In a more recent work in 2006, Qiao et al. [181] utilized handwriting graph representation and attempted to determine the smoothest path from the graph. The researchers reported a restoration accuracy of 96% for a single-stroked handwriting image taken from the Unipen [71] handwriting database. Chouinard and Plamondon [34] approached the problem with the line following scheme. This approach reported significant distortion reduction at the intersections. Despite the high speed advantage, this technique has a drawback as it may generate double lines at intersections. However, the authors suggested that another thinning algorithm could be employed locally to remove such lines, if necessary.

There is an unclear boundary between preprocessing and feature extraction. Some final steps in preprocessing can be considered as part of the feature extraction process and vice versa. The products of preprocessing can be external boundaries, contour boundaries, and skeleton representations.

### 2.6 Features Extraction

The process in which digital information is modified, simplified, and combined so that the salient information can be processed and classified, is called feature extraction. Ideally, a feature extraction technique must be justifiable by the set of rules that govern the formation of the class of pattern being considered and produce no classification error. However, the production of signatures is too complicated and involves a large number of degrees of freedom. As a result, feature extraction techniques commonly proposed for signature verification in the literature have no supportive rule and their performance is all determined experimentally.

From the perspective of questioned document examiners, Hubert and Headrick [86] suggested that an automatic signature verification system must also consider letter design, line quality, stroke sequence, pressure variance, line continuity. To the best of our best knowledge, none of the proposed automated approaches meets the above requirements. Moreover, the proposed approaches are found to be lacking a mechanism for justifying its decision on the authenticity of the signature being examined. There are several types of forgeries that may be categorized by their nature and level of sophistication.

Feature extraction techniques are crucial to the success of automated pattern recognition systems [14]. They extract essential information from input images and represent this information in a suitable format for the learning process and the decision process. Good features are those that enable the system to identify the correct classes that patterns belong to with the least amount of errors. For the signature verification problem, the distribution of feature vectors extracted from genuine signatures should also be denser than their forgery counterparts to reflect the fact that intra-personal variations are not as high as interpersonal variations. Baltzakis and Papamarkos [14] commented that the selection of features must be appropriate for the application and the approach. Klement et al. [114] summarized the three requirements that concerned the feature selection process: (1) Speciality (Minimizing intra-class variability and maximizing inter-class variability); (2) Universality (can be applied to any writer); (3) environment independence (with respect to writing instruments and materials). In other words, it is essential that a feature extraction technique could
minimize or even eliminate the negative effects from variations such as rotation, shift, or dilation of the pattern being considered.

Different cultures have different writing styles and this writing style would affect the accuracy of feature extraction methods. For example, a curvature based feature may be effective with cursive script and continuous scripts but may have little effect on scripts like Japanese, Chinese, or Korean which consist of discrete straight strokes. Moreover, Eastern signatures usually consist of more than one character and can be written vertically, which is different to Western signatures. As a consequence, many feature extraction techniques specialized for Chinese or Japanese signature verification cannot be applied directly to Western signatures.

In a survey by Plamondon and Lorette [174], the authors noticed that no static feature extraction technique was capable of signature reconstruction. This implied that the static techniques developed at that stage may not have captured enough salient information, or, salient information could not be captured with static techniques. If this could be proved, the former option would likely have put a full stop to the off-line branch of signature verification. With the increasing power of computing, researchers are now able to investigate more sophisticated off-line feature extraction techniques. Justino et al. [98] employed a large grid based feature with size up to $63 \times 10$. With such a large feature vector, consisting of 2520 features, to some extent, a signature image could be reconstructed. However, the reported results are still far from that of their on-line counterparts. This is where the question of whether the information should be treated as discrete pieces arises.

Features can be categorized based on the scope of the area of interest to be Global Features or Local Features, as summarized by Hou et al. [79]. Researchers believed that an appropriate combination of global and local features would improve classifiers’ robustness in distinguishing forgeries and tolerating intra-personal variances [88, 79]. Global features treat a pattern as a whole and tend to be less sensitive to variation or noise compared to local features [180]. Commonly investigated global features include the following: aspect ratio [14, 10, 132], geometric centre or centre of gravity (COG), slant angle, image area. Local features are those having elements extracted from a small and limited region of the pattern. They provide the system with more information and with higher details of the pattern. As a compensation for computational cost and vulnerability to noise as well as variations, local features are more informative than global features. Some examples of local features are “Grid segmentation” proposed by Justino et al. [99, 97, 98], “Box method” by Hanmandlu et al. [75, 138], and the Modified Direction Feature [23].

Researchers also classify features as being macro or micro depending on the object being considered. At the character level, Leedham and Pervouchine [128] considered line and word spacing, line slant as being macro-features whilst the loop size of a character, ascender/descender length, ligature length between characters, etc. were micro-features. According to these researchers, micro-features are endowed with individual traits and are more difficult to forge than macro-features.

Features can also be considered to be static or dynamic. As previously mentioned in Section 1.2, static features are those related to the spatial distribution of the pixels of the handwriting static images whilst dynamic features are those captured on-the-fly with reference to temporal information. Dynamic features may also refer to dynamic information of the writing instruments at certain time or location. Some researchers attempted to recover dynamic information, such as axial slant, pressure, etc., at each point on the trajectory of the handwriting from the grey-scale image of a signature in higher resolution [9, 38, 8]. Sometimes, such features are called Pseudo-dynamic Features. It is obvious that the extraction of pseudo-dynamic feature is significantly affected by writing instruments. Hou et al. [79] observed that static features are usually used to
identify Simple forgeries whilst Pseudo-dynamic features can be used to detect skilled forgeries. This opinion is shared by Ferrer et al. [57] who supposed that the static approaches are suitable for random and simple forgeries whilst the dynamic or the pseudo-dynamic approaches would be more appropriate for skilled forgeries.

2.6.1 Grid Based and Tree Based Features

Grid segmentation has been adopted by many researchers for feature extraction [83, 100, 99, 97, 14, 217, 194, 98, 202, 121]. The size of grid elements can be uniform or flexible. In the uniform segmentation scheme, the signature image is first zoned by vertical and/or horizontal lines with equal distance. From each sub-image obtained, one or more global features are extracted. Justino et al. [98] employed a \( 63 \times 10 \) grid to extract up to 2520 features in total for each input signature image. In that work, the global features extracted from each cell were the Pixel Density, Gravity Centre distance, Segment Curvature, and Predominant Slant. The authors also reported that a very high verification rate of 96% could be obtained with the SVM classifiers on a signature corpus consisting of 100 individuals. They noticed that under similar experimental parameters, except for the number of genuine signatures for training, the classifier should learn more about the intra-personal variations. As a result, the FRR tends to be lower whilst the FAR for targeted forgeries tend to increase. This trend is supported by Shih-Yin et al. [198] who reported an increment in EER of 0.5% for skilled forgeries when the 4 genuine signature were employed for training instead of 2. The above observation contradicts the observation of Nguyen et al. [153], and Ferrer et al. [57]. The difference may arise from many sources such as database, feature extraction technique, methodology, etc., and requires further investigation.

Larkins and Mayo [121] argued that the uniform approach does not capture the same structural properties of corresponding regions among signatures belonging to the same writer. They proposed that an adaptive size grid based on the number of black pixels should be employed instead. The distance between two consecutive grid lines, vertical or horizontal, is determined using the respective histogram. This innovative approach, to some extent, reduces the undesired effects, which may occur from vertical and horizontal local shifting. A similar approach was employed extensively by [102] and Srihari et al. [202]. In their work, the signature images were divided horizontally and vertically into rows and columns having the number of black pixels in every row or column being equal. After that, subsequent features such as Gradient, Structural information, and Concavity are extracted from each rectangular box.

The feature values extracted with the grid-based approach are prone to be sensitive to all types of variation, especially when the adopted grid is of high density. As a result, the grid segmentation approach has the potential to effectively reject forged signatures since it can distinguish minor variations. Despite the promising results reported, there has been no in-depth research in the literature on how to select an optimal grid size.

Tree-based feature extraction techniques are relatively more sophisticated than their grid-based counterparts. This image segmentation scheme successively partitions the image of a pattern using points with special characteristics. Jena et al. [93] employed the geometric centre and successively determined 60 geometric points from the signature image. Using a similar scheme, Afsardoost et al. [2] extracted four geometric centres from the signature contour. The recursive nature of tree-based feature extraction techniques makes them seem to be more effective in capturing the characteristics of fractal patterns rather than signatures. Nevertheless, encouraging verification
2.6.2 Rotation Invariant Features

During the acquisition process, signature images may not have been properly aligned. Other physical conditions such as inconvenient writing posture may also create rotation, with respect to normal writing conditions in signatures. Some researchers believed that the accuracies of a recognition system could be improved by employing features which are invariant to rotation [221]. A feature extraction technique is considered to be immune to rotation if its feature values remain relatively unchanged whenever the input pattern is rotated. As previously mentioned in Section 2.5, an approach to this problem is rotating the input pattern by a correction angle. Nevertheless, the choice of an appropriate correction angle remains an unsolved problem, and rotation invariant features still attract the attention of researchers.

As can be found in the literature, many of the feature extraction techniques are rotation invariant. For example, in their signature verification research using neural networks (NNs) in conjunction with a Euclidean distance classifier, Papamarkos and Baltzakis [161, 14] employed the centre of mass, image area, number of closed loops, number of edge points, number of cross points, and texture which are all rotation invariant. The majority of these features were extracted from the skeletons of the signature. These researchers reported verification accuracies up to 81.81% produced with their proposed verification system.

Rotation invariant features usually belong to one of the following categories: all angles with or without a fixed origin and curvature-based. In the all angles approach, the input patterns are examined from all angles. To implement a fixed origin feature, a point must be chosen and its relationship with other points are considered. Finally, a normalization process is usually employed to eliminate the phase shifting effects. Jing et al. commented that the origin itself should be rotation invariant, e.g. the geometric centre or the centre of gravity. In their work [95], these researchers employed the COG of the signature as the origin of the coordinate. A circle whose centre is the COG and encloses the signature, is then placed over the signature. Next, the distance profile between each point on it to the signature’s external boundary on the line connecting it and the COG is determined. This profile is called the Ring-External-Feature (REF). The second profile, which is called the Ring-Internal-Feature (RIF), deals with the distance between the COG and the first points of the signature on the line from the COG to the circle. In the final steps, the Fast Fourier Transformation (FFT) is employed to eliminate the phase-shift effect. Experiments using this feature extraction technique in conjunction with the Mahalanobis distance classifier over a private database of 55 writers produced a promising AER of 10.4%. However, the comparability and practicality of this result is relatively low. The training process in this research employed a large number of genuine signatures (23 – the leave-one-out method). One drawback of the above approach is that the selection of the origin is unjustifiable. There has been no theory or research describing the relationship between the signatures and any of the proposed origins.

Coetzee et al. [36] demonstrated that the origin selection problem can be avoided by employing ring-topology HMMs in conjunction with an appropriate all degrees feature extraction technique. Each state of their proposed HMM topology has two adjacent destination states and two adjacent source states. The states are also equipped with a state skip. Figure 2.6.1 illustrates a ring HMM. In their research, the HMMs were trained and tested using feature vectors extracted using
a Discrete Radon Transformation (DRT):

\[
R(p, \theta) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} g(x, y) \delta(p - x \cos \theta - y \sin \theta) \, dx \, dy
\]  

(2.6.1)

where $R(p, \theta)$ is the accumulation of points shared by the pattern being considered with the line parametrised by slant angle $\theta$ and distance to the origin $p$; $g(x, y)$ is the image intensity at point $(x, y)$; and $\delta(r)$ is the Dirac function. The ring HMM topology also keeps the proposed system from the problem of the phase-shifting effect. Their system is reported to have outperformed Dolling’s algorithm on the same database.

In other research by Sanchez [192], Martinez [139], and Ferrer et al. [57], the features were extracted from the envelope description and the interior stroke distribution of a signature in polar and Cartesian coordinates. In their techniques, the geometric centre was chosen as the origin of the coordinate. Ferrer et al. [57] reported a rejection rate of 87.4% for targeted forgery (12.6% FAR) with the HMM classifier [57]. Despite the encouraging results, the choice of the geometric centre as the origin coordinate with regards to the formation of the signature remained unjustified in their research.

Feature extraction techniques based on the Hough [50] and Radon Transforms can also be categorized as being fixed origin features. Originally, the Hough Transform and its variations are employed in detecting different types of patterns such as straight lines, circles, and ellipses [142]. In their research in signature recognition, Kaewkongka et al. [101] employed the Hough Transform for detecting straight lines to map signature skeletons into Hough space. The recognition process was performed using a back-propagation neural network. Researchers reported an encouraging recognition rate of 95% with a modest collection of 70 signatures. Coetzer et al. [36] investigated the application of the Discrete Radon Transform in conjunction with Hidden Markov Models for the off-line signature verification problem. Employing a similar approach enhanced by biometric strengthening, Shih-Yin et al. [198] reported the EERs of 1.1%, 1.2%, and 2.1% for random, casual,
and skilled forgeries respectively.

Rotation invariance can also be obtained using curvature information extracted from the contours. As early as 1977, Freeman [61] introduced the line-segment scan method to identify curvature discontinuities, and critical-points. One notable characteristic of curvature is that this feature value is invariant when the object’s location is shifted or rotated. Besides, a curvature-based feature does not require a fixed origin whose selection could introduce more bias. It is evident that any minor changes in the trajectory, which is often seen in forgeries, can be easily detected in the curvature profile by direct comparison using reference signatures. Thus, curvature-based features have been investigated in pattern recognition and signature verification [39, 224, 229].

Despite the promising applications, the numerical calculation of curvature is not easy due to the involvement of the first and second derivatives of points. The mathematical definition of curvature at point \( t \), having coordinates \((x'(t), y'(t))\) is:

\[
K(t) = \frac{(x'(t)y''(t) - y'(t)x''(t))}{\sqrt{(x'(t)^2 + y'(t)^2)^3}}
\]  

(2.6.2)

According to [65], when the curvature is given implicitly by \( g(x, y) = 0 \) then the curvature is

\[
K = \frac{g_{xx}g_y^2 - 2g_{xy}g_xg_y + g_{yy}g_x^2}{(g_y^2 + g_x^2)^{3/2}}
\]  

(2.6.3)

The calculation of curvature from discrete approximate boundaries points and their second derivatives is a non-trivial task. Dealing with this problem, smoothing filters such as high-order B-splines [224], or a Gaussian filter [229] can be employed. However, if the pattern is complicated and consists of several elements and such elements are not always connected, the curvature profiles extracted from such patterns can be totally different. In their research [39], Deng et al. approached this problem by applying a \( 3 \times 3 \) morphological mask to dilate the contours. Similarly, this trick was employed in the more recent work by Zhang et al.[229].

Once the boundaries of the specimen are normalized and the curvature profile is extracted, the next step is determining the start phase of the periodic signals of signature contours. From the contours of the signature image being examined, there can be several one-dimensional periodic signals for extraction. As these signals are periodic, it is necessary to choose a start point for each signal. Yang et al. [224] proposed a solution to this problem with a new concept of a distribution centre.

Zhang et al. [229] suggested that the verification of off-line signatures can be performed using their envelope solely and investigated the modelling of signatures using curvature information. However, the curvature profile of a signature may distort significantly due to intra-personal variation. The shift of a single horizontal main stroke may cause great variation in curvature profile. To overcome this obstacle and to make the curvature extracted more stable, researchers chose to dilate signatures before extracting their envelope. Using the proposed feature extraction method, these researchers reported that an AER of 17.17% was obtained for a database of 80 signature sets each consisting of 24 genuine signatures and 30 forged signatures. Another approach to minimize the effects from such variations is to employ the trajectory instead of the boundary. However, the recovery of trajectory from off-line handwriting images is not a trivial problem.
2.6.3 Other Features

Inspired by the results that dynamic time warping (DTW) obtained in speech recognition, Shanker et al. [167] investigated the application of DTW in the area of off-line signature verification. The authors believed that, the ability of DTW to optimally stretch or compress to match two signal sequences could be employed to compare the similarity between two signatures. Horizontal and vertical projections of the signature were employed to provide two sequences of signals for similarity comparison. The optimal match profile is then further segmented and a score is given for the similarity of each segment and its respective partner. With this modification, an AER of 22.5% was reported. It seems that the divide-to-conquer strategies helped improve the verification accuracy.

From the exterior boundary of signature images, Bansal et al. [15] attempted to extracted critical points using the poly-fit function. In their research, the external boundaries of signatures were employed to construct a writer signature model. Despite the fail to enrol rate (FTE) of 12%, the researchers reported relatively low FRR and FAR rates of 2.64% and 13.02% respectively, whilst using only 4 genuine specimens in the model construction process.

2.6.4 Feature Vector Dimension Reduction

Another issue concerning feature extraction techniques is reducing the dimension of the feature vector. Large dimension feature vectors do not always associate with better results, but always consume more computational resources. Consequently, it is necessary to keep the size of feature vectors small whilst retaining the most discriminative information.

In many local feature extraction techniques, the dimension of the feature is usually controlled by the number of “local areas”. Feature size reduction can be performed by simply employing less local areas. However, there exists a trade-off between accuracy and feature vector dimension.

One of the most simple dimension reduction techniques is local averaging. This technique replaces a set of numbers by their average value so that the dimension is reduced. In their prominent cursive character recognition research, Blumenstein et al. [24] grouped adjacent location and directions at transition values together and represented the grouped values by their averaged values to form the MDF feature.

Compared to local averaging and other image filtering techniques, Principal Component Analysis is more sophisticated. Given a set of vectors of a vector space \( \mathcal{D} \), PCA gradually finds the largest eigenvectors and corresponding eigenvalues of the covariance matrix of input data. The projection of data on the new coordinates containing largest eigenvectors reveals the most or the least significant variance. In pattern recognition, especially handwritten digit recognition where the number of training samples is relatively large, dimension reduction using PCA has been investigated [227, 196]. However, the nature of the PCA does not guarantee an improvement of class separability. Originally, PCA is only capable of finding linear subspaces and is thus inappropriate for the handling of non-linear manifolds. Besides, researchers noted that there has been no detailed analysis or method to calculate the optimal number of principal components to be kept [33].

2.6.5 The Combination and Selection of Features

It is a well known fact that better accuracies can be obtained by combining two or more features. However, putting features together does not necessarily produce better results. In many cases, the
results obtained are often unsatisfying. As a consequence, researchers have investigated methods to search efficiently for better feature combinations.

Providing a training database with genuine signatures of $N$ authors, Aguilar [3] proposed a feature ranking method using inter-user class separability based on the Mahalanobis distance:

$$S(F_k) = \sum_{i=1}^{N} \sum_{j=1}^{N} |d_{i,F_k}^M - d_{j,F_k}^M|$$

(2.6.4)

where $d_{i,F_k}^M$ is the Mahalanobis distance between the mean of the $F_k$-parametrised author training signatures of author $i$ and the $F_k$-parametrised set of all training signatures from all authors. The advantage of this scheme is simplicity. However, it does not take the interaction between two or more features into account.

An exhaustive search is feasible only if the number of features is relatively small. As the number of combinations grows exponentially with additional features, alternative optimisation search techniques, such as genetic algorithms, must be employed. In the Genetic Algorithm (GA), the development through generations of a population of candidate solutions is simulated. The interaction between members of the population are driven by predefined probabilistic genetic operators and objective functions. Pervouchine and Leedham [165] noted that GAs are capable of finding near optimal solutions relatively quickly, whilst they are unlikely to get stuck in a local extrema compared to gradient-based search techniques. Besides, the performance of GAs is not very much influenced by the values of parameters. Solutions can still be reached even if these values are far from optimal with a larger number of generations.

In recent work, the performance of GAs in on-line signature verification was investigated by Galbally et al. [64]. These authors compared the performance of GAs to the feature ranking technique using the Mahalanobis distance and concluded in favour of GAs. In their experiments, the results clearly indicated the superiority of GAs over the ranking scheme. Depending on the number of features employed, the improvements could be from 1% to more than 4%. These researchers also noted that the dynamic features based on speed and acceleration are the most appropriate features for the identification of skilled forgeries.

Another notable research using GAs for off-line signature verification is that of Bertolini et al. [19]. In that work, the chromosome is an 84-bit string representing the activation state of 4 local features extracted using 21 grid settings. Each bit indicates whether to include a local feature (e.g. distribution, curvature, density, or slant) extracted at a particular grid dimension. The genetic operations were one-point crossover, bit-flip mutation, and roulette wheel selection. Interestingly, a convergence of performance of various objective functions was observed with the increasing size of the reference set.

The major disadvantage of GAs is the extremely large number of evaluations required. The training and classification processes must be performed in order to obtain the performance of every member of the population created by the GAs. This restricts GAs from employing sophisticated classification techniques such as neural networks, SVMs, HMMs whose computations are relatively expensive, although it does not rule out their use entirely.

### 2.7 Trajectory Recovery

The research in automatic handwriting recognition has been intensively pursued for nearly four decades and has obtained some significant achievements [176]. Successful applications include:
postal address recognition, handwriting recognition and signature verification using tablets, historical document recognition, and form processing.

As previously discussed, automatic recognition systems can be categorized as being on-line or off-line based on the availability of dynamic information. On-line recognition is usually performed using temporal spatial information generated from the movement of a stylus on the surface of an electrostatic or electromagnetic tablet. Depending on the hardware, this signal stream of information may include: pen-inclination, pressure, velocity, acceleration, movement direction, number of strokes. From such information, the corresponding static image could be simulated \cite{181} using ink deposition models in conjunction with trajectory interpolation functions.

Unlike its on-line counterpart, off-line recognition employs only the static images captured by optical devices such as a camera or scanner. Due to the absence of dynamic information, the accuracies of off-line recognition systems could not be as high as on-line recognition \cite{176}. As a trade-off, the on-the-fly collection of dynamic information restricts the applications of on-line recognition and gives off-line recognition certain unique advantages such as the ability to capture information remotely and conveniently.

The success of on-line systems \cite{176} encourages the recovery and utilisation of dynamic information, such as pressure \cite{8} and, especially, stroke order to improve the performance of off-line recognition systems. Research in the field of psychology also suggests that the human perception of dynamic information from static images assists in the recognition of characters \cite{13}. It is strongly believed that if the trajectories are properly recovered, the performance of automatic off-line handwriting recognition systems could significantly be improved \cite{175, 219, 45, 30}. Handler \textit{et al.} \cite{74} reported a recognition performance downgrade when off-line data was simulated using on-line data. Experimental results from \cite{219, 118} later confirmed that the time ordering of the signal contains important information for the recognition of handwriting. In the literature, time ordering information has been used for word segmentation and recognition \cite{175, 118}, character recognition \cite{131}, numeral recognition \cite{125, 127}, and writer identification \cite{166}. The application of recovered trajectories in signature verification has also been investigated. In \cite{163, 182, 180, 233}, on-line data which was previously obtained in a registration process provided valuable clues for the recovery of signature trajectory. In another attempt to extract and utilise the temporal order, Munich and Perona \cite{144} tracked the signature images as writers signed with the assistance of a camera. Despite the large number of potential applications, trajectory recovery remains an open problem.

There are two major processes in a trajectory recovery system: local examination and global reconstruction. Local examination provides the essential information which will be referred to in the global reconstruction phase. This often includes the detection and analysis of junctions or ambiguous zones, endpoints, double-traced lines but can also be extended to grey level consistency, striations, feathering, pressures, and accelerations \cite{47}. In global reconstruction, the overall trajectory is determined using the information obtained from local examination. The outcome of this process can be a list of ranked trajectory candidates which may further be analysed using a knowledge-based module \cite{219}. The main components of a trajectory recovery system are illustrated in Figure 2.7.1.

The remainder of this subsection is organized as follows: The next section, Section 2.7.1, presents both the advantages and disadvantages of input material, the skeleton and the contour. Section 2.7.2 briefly mentions the preprocessing of input images. Local examination is then detailed in Section 2.7.3. Section 2.7.5 is devoted to global reconstruction techniques. Finally, issues
concerning performance evaluation are discussed in Section 2.7.6.

2.7.1 Material

Trajectory recovery systems found in the literature employ either the skeleton or the contour of the handwriting [175]. It is believed that a successful trajectory recovery would take advantage of and utilise both the skeleton and the contour.

2.7.1.1 Skeleton

Pioneering work in handwriting trajectory recovery [125, 127, 160, 126] employed 1-pixel line width thinned images, which is often called the skeleton. The main advantage of skeleton-based approaches is their computational efficiency, whilst maintaining acceptable geometric and topological attributes [119].

Ideally, the skeleton should be identical to the original pen tip trajectory [166]. However, traditional thinning methods often produce artefacts, such as bifurcations or elongations and represent original blobs and filled holes by lines instead of loops. Such anomalies incorrectly describe the structure of the source pattern and make the recovery more difficult [127, 85]. Besides, skeletonisation is considered to be highly sensitive to noise [175]. A single isolated background pixel could result in a loop being created. In reality, some researchers [85, 129, 91] exclude unreliable skeleton segments from consideration whilst others treat those as clues and examine their internal structure carefully [181, 107]. Researchers have also investigated specialized handwriting skeletonisation techniques [166, 150, 120, 37, 206, 108].

In [166], the skeletons were interpolated from selected points using B-Splines. The junction spline knots were shared amongst all the incoming branches before being decomposed in separate knots after a fine tuning process. Consequently, the structure of the characters was preserved. Spline knots have also been investigated in [110] where the control points were over-generated equidistantly before being selectively removed to obtain the optimal set of control points. These pseudo-skeletonisation techniques were reported to be less sensitive to noise compared to traditional thinning methods.

Despite the ongoing investigation using the skeleton, some researchers suggested that the temporal information cannot be recovered from the skeleton using heuristic rules only [47]. It is believed that each clue about the motion of the writing instrument should be carefully examined in order to recover the trajectories successfully.
2.7.1.2 Contour

Another aspect for trajectory recovery is the handwriting contour. Compared to the skeleton, the handwriting contour does not frequently contain anomalies. Each point on the contour corresponds to a position of the pen tip and is a clue for recovery.

In Plamondon and Privitera’s research [175], the contour was employed to recover the trajectory of handwritten words. Initially, curvature local maxima points were employed to locate ambiguous zones. Later on, two branches of a crossing were joined together based on contour curvature smoothness. The proposed system was tuned using a number of databases and the performance was evaluated using an untouched database consisting of 200 words, which were written by six writers. The successful ambiguous zone interpretation rate was reported to be 94% whilst the original pen tip movement recovered was 89%.

In Doermann et al.’s research [44], the handwriting contour has been employed to locate and recover hidden loops in three phases. Firstly, candidate contour segments are located. Secondly, candidates that a-priori do not have an elliptic shape or are not surrounded by a visible loop are discarded. Finally, blobs that meet the elliptic shape requirement are selected. In another work on loop recovery by Steinherz et al. [205], the contour was used to classify holes, identify hidden loops and hidden natural sub-loops.

2.7.2 Preprocessing

Similar to many other handwriting recognition problems, the performance of trajectory recovery systems are considerably affected by the quality of the handwriting static image. Therefore preprocessing is necessary to stabilise the quality of the input image [176] prior to any further analysis (Figure \ref{fig:system}). This process may include, but is not limited to, grey-scale conversion, binarisation, noise removal, broken stroke restoration [199], and contour smoothing.

Conversely, some researchers argue that certain preprocessing techniques reduce the robustness of a trajectory recovery system [111]. A popular preprocessing operation such as binarisation can deteriorate or even destroy valuable clues such as intensity consistency, continuation, and feathering which could possibly be extracted and utilized using grey-scale images [46]. As a result, grey-scale based trajectory recovery techniques have been investigated [111, 109, 164].

2.7.3 Local Examination

Doermann [47] suggested that trajectory recovery techniques can produce useful information when obtained from stroke and sub-stroke features as well as knowledge about the writing process. The detailed taxonomy of temporal clues can be found in [47]. Despite the promising outcomes, Plamondon and Privitera [175] noted that the examination of handwriting at the sub-stroke level would be computationally expensive. Local examination includes the following sub-processes: ambiguous zone detection, ambiguous zone analysis, double traced writing and hidden loops analysis, the detection and pairing of endpoints.

2.7.3.1 Ambiguous Zone Detection

There are parts of the writing where the establishment of writing order is not straightforward. Many of those are occluded i.e. start/end points, crossings, and touching components. They
are often named ambiguities or ambiguous zones. To recover the intrinsic trajectory of the pen movement from a static image, these ambiguities must all be located and analysed.

In skeleton-based approaches, the detection of ambiguous zones often relies on the average stroke width [181, 85]. At these ambiguous zones, the distance from a skeletal point to the nearest background pixel appears to be larger than half the line width. Since line width often varies with writing instruments and writing speed [46], especially in sophisticated handwriting or signatures, handling ambiguous zones this way requires greater care.

In [175], Plamondon and Privitera demonstrated that ambiguous zones can be located using local curvature maxima points of the handwriting contour. According to the authors, curvature maxima of the contour correspond to either the overlap of two consecutive motor strokes or two distinct strokes. Similarly, Zhong-sheng et al. [230] classified handwritten Chinese ambiguous character zones into basic and complex types using discontinuous points [72]. However, the problem of curvature estimation for discrete points itself is not an easy problem [77, 223].

In research into Arabic handwriting trajectory recovery, El-Baati et al. [51] demonstrated that ambiguous zone detection could be performed in a straightforward way. Their technique employs a square window sweeping through the image and counting the number of background regions parted by the foreground. 3 background regions correspond to a ‘Y’ branch point whilst 4 means an ‘X’ crossing.

2.7.3.2 Ambiguous Zone Analysis

The analysis and matching of the incoming and the outgoing segments branching from ambiguous zones can be considered one of the most crucial and challenging tasks in trajectory recovery [181]. This operation often involves the evaluation of continuity or smoothness for each pair of lines that branch out from the ambiguous zone.

In [127], ambiguous zones in the signatures were analysed using heuristic rules. A signature is then represented by a set of critical points extracted from the recovered trajectory. From this work, a recognition rate of 97% was reported. In [85], two branches are joined if the magnitude of direction variation is smaller than a given threshold. Heuristic rules were also proposed to resolve situations where thresholding failed. A similar approach has also been employed in [90, 91]. In addition to direction, stroke width and length have also been employed to evaluate continuity in [27]. Curvature based continuity functions using Kalman [116], Gaussian [175], and B-Spline fitting [146, 145] have also been investigated. Nevertheless, grey level consistency is considered helpful for this task [46].

Qiao et al. [181] surveyed that 95.8% of all the intersection areas are of degree 4 (joining 4 branches) and 95.1% of those having degree 4 are crossing nodes. This implies that proper analysis of degree 4 intersections would significantly contribute to the overall accuracy. Neural networks have been proposed to distinguish crossing type from other types based on the tangential direction of branches. The tracing through ambiguous zones would also be assisted by the in depth analyses of the skeletal structure.

When the degree of an intersection was small, it was able to list all the topologies for the crossings. In his research, L’Homer [129] noted that this prior knowledge significantly assisted the analysis of the ambiguous zone as well as double traced segments.
2.7.3.3 Double Traced Writing and Hidden Loop Analysis

Double traced writing can be defined as a segment of writing which is formed by two consecutive strokes terminating and starting in near opposite directions. A loop, which is made of several strokes, is formed when the writing instrument revisits a previous position while touching the writing surface continuously [205]. Depending on the nature of the writing instrument, a small loop may collapse to a blob, namely a hidden loop. Within context, humans can identify these artefacts relatively easily. Such segments also help distinguish one letter from another [44]. In a trajectory recovery system, the analysis of double traced writing and hidden loops is often performed after the ambiguous zones have been detected and analysed.

In [107], Kato and Yasuhara provided the taxonomy of double traced writing (D-line) which includes looped (L), proper (P), and spurious (S) D-lines. The angles between branches and writing behaviour were analysed to heuristically detect D-Lines in this research. Similarly, angles between branches were used to construct the weighted matrix of a general graph maximum weighted matching algorithm whose best solution would highlight the D-lines [181].

Intuitively, Abuhaiba et al. [1] suggested that the points lying deep inside the blob are more likely to belong to the background. The distance to the contour threshold was set to be the distance between the majority of the skeleton pixels and the contour plus 2. Explaining the modest recovery rate of 83.6%, the authors commented that both line width and blob size vary even in the same stroke that caused the introduction of spurious holes, which negatively affected the performance of their recognition system. This view is also shared by Doermann et al. [44] who later noted the impracticality of this technique due to the high signal to noise ratio.

In their research, Doermann et al. [44] examined the blobs using the mutual distance measurements between the two sides of a symmetric shape. According to these researchers, a blob often resembles an ellipse. Therefore, after contour partitioning and the selection processes, only blobs that resemble elliptic shapes are considered in the shape analysis process.

In another hidden loop recovery study using the contour, Steinherz et al. [205] used contour bank segments to determine the next course of the pen trajectory. Their approach is to first divide the contour into smaller segments, or atomic entities, using discontinuity points. After that, the relationships between atomic entities are established using heuristic rules. A relationship between two atomic entities can be correspondent, continual, or discontinual. The identification of hidden loops relies on these relationships.

2.7.4 End Points Analysis

After all the ambiguous zones have been detected and analysed, all pairs of corresponding start and end points need to be identified before the trajectory can be traced globally. This essential process usually includes the identification of stroke ends and hidden ends, selection of beginning points, merging of broken strokes resulting from preprocessing. Figure 2.7.2 illustrates a handwritten word
where both start and end points are occluded.

It is agreed that the position of stroke ends largely depends on writing styles. For a recovery system to be successful, such knowledge should be referred to. In Arabic handwriting, branch points and end points are always located to the left of the starting point [51]. For a Latin right-handed writer, the writing usually begins from the top left and progresses downwards to the right [86]. Rousseau et al. [188] investigated the significance of knowledge inherent in handwritten Latin letters in recovering the writing trajectory. The conclusion was that prior knowledge produces improved recognition rates.

As stroke ends may be hidden or occluded by other strokes, especially in isolated characters, the identification of the start points and the end points is not trivial and sometimes impossible. Many researchers have chosen to exclude patterns with hidden ends to simplify the problem [107, 85, 122, 187]. In Rousseau et al.’s research [187], as many as 9% of the isolated character samples were reported to have at least one hidden end and were removed from the experiments. The detection and analysis of end points is even more challenging in multi-stroke handwriting and signatures, which vary greater in style and contain a larger number of ambiguous zones. These facts partly explain why it is observed that there has not been any attempt in the literature devoted to the automatic hidden end identification problem [187].

2.7.5 Global Reconstruction

The final process of a handwriting trajectory recovery system is to put every recovered segment and stroke back into correct writing order, namely global reconstruction. This is necessary since the handwriting may consist of more than one pair of pen-down and pen-up events. Moreover, there may still be ambiguities left that cannot be analysed and only global reconstruction can enumerate all possibilities. In this process, the direction of pen movement in each segment is also established.

In their research, Bunke et al. [30] applied the best-first search technique on weighted graphs to find the optimal trajectory. The search graph was constructed from the skeleton of characters and the costs were computed with the consideration of the writing direction, path minimization, continuity, and direction of the loop.

In the global graph search approach, the topological structure of the handwriting image is described by a graph. The end points, junctions, and touching points are represented by vertices and the lines together with curves are represented by edges. The pen trajectory is finally determined by finding the most appropriate path which traverses every edge exactly once. In Jäger’s research [91, 89], each handwriting segment is represented by a vertex of a weighted graph. Two vertices are connected by an edge if the two corresponding segments join the same junction. The deviation between this pair of vertices is then assigned to this edge. The final trajectory is then determined by finding the Hamilton path which minimizes the curvature cost, and which may lead to a combinatorial explosion [106].

In another work concerning single stroke handwriting where the degree of crossings could be equal or less than four [107], Kato and Yasuhara managed to detect double-traced edges and simplified the global reconstruction process to the task of searching for an Eulerian cycle from a directed graph. This technique has also been employed and validated in Rousseau et al.’s research [187] on isolated letter trajectory recovery. The lower degree of crossing constraints also enabled El-Baati et al. [52] to breakdown Arabic handwritten words into segments and utilise a genetic
algorithm to search for the optimal trajectory.

The recovery of handwriting trajectory can also be assisted with dynamic information for recovering handwriting partially [231, 232, 233] or globally [148, 149, 151]. In [148], the trajectories were considered as sequences of position and direction variations. Hidden Markov Models (HMMs) were adopted to represent this information extracted from the skeletons of static signature images. The state sequences are then determined by matching the HMM to dynamic exemplars using the Viterbi algorithm [184, 185]. The optimal state sequence is finally selected by comparing the likeliness between the HMMs and its corresponding dynamic sequence exemplar.

### 2.7.6 Performance Evaluation

It is apparent that the performance of trajectory recovery techniques is subject to experimental settings. To determine the performance of a trajectory recovery system, the recovered trajectories need to be compared with the online ground truth. This can be performed by either using dual databases which consist of simultaneously generated on-line and off-line data. Another approach is to synthesize off-line images using the on-line data [181] and an ink-deposition model such as the one described in [59]. Morphological operators such as bicubic interpolation and anti-aliasing may be employed in latter stages to obtain smoother writing [110].

As demonstrated in the literature, the performance of some trajectory recovery techniques was determined visually [52] or reported indirectly, i.e. recognition accuracy. Visual performance evaluation is feasible only if the number of testing samples is relatively small [107, 27]. Moreover, visual evaluation is subjective, not quantitative, and is prone to error. Some automatic performance evaluation protocols have been proposed to overcome such limitations.

In Niels and Vuurpijl’s research [156], dynamic time warping (DTW), an elastic matching technique, was employed to match the recovered trajectory and the trajectory traced by handwriting experts. In [123], Lau et al. replaced Kendall’s distance in the Feigin and Cohen ranking analysis model by a connection metric for performance analysis. This metric takes into account stroke direction and stroke connection. In more recent research using signatures, Nel et al. [151] constructed Hidden Markov Models for the purpose of performance evaluation from online exemplars. These HMMs are capable of identifying insertion, deletion as well as substitution errors quantitatively.

### 2.8 Classification Techniques

The detection of random and simple forgeries can be regarded as a signature identification problem which is much easier. Many researchers report the FRR for random forgeries as low as 0.1%. Sansone and Vento [193] reported the FRRs of 0.01% and 4.29% for random and simple forgeries respectively over a database consisting of 49 signature sets.

The following subsections are devoted to off-line skilled forgeries detection. Major approaches to the automatic signature verification problem are the Fuzzy Model Approaches, Statistical Learning Approaches, and Distribution of Distances.

#### 2.8.1 Distance Classifiers

The advantages of distances classifiers over other approaches such as statistical learning classifiers are their simplicity and speed. Popular distance classifiers are Euclidian and Mahalanobis distance classifiers. In this approach, the sequence of the signature model is obtained by averaging the
training sequences extracted from genuine specimens. In order to verify a questioned signature, the system calculates the mean Euclidean distance between the input sequence and the model sequence. The authenticity of the signature is judged by comparing the distance obtained with a threshold [57].

In their research, which employed geometric parameters of the signature, Ferrer et al. [57] reported that the performance of Euclidian classifiers were approximately 2% worse than SVMs. Considering the fact that the implementation of this classifier is much simpler than the SVM and can be deployed in a smart card, this result is very promising.

Fang et al. [55] utilized variations in relative stroke positions in two-dimensional signature patterns to construct the writers’ models. Questioned signatures were directly matched with those of genuine specimens using a two-dimensional elastic matching algorithm. This process resulted in a displacement vector, which was then multiplied by the covariance matrix of training vectors to achieve the signature score. The decision on the authenticity of a questioned signature was made by comparing this score with a given threshold using the Mahalanobis distance.

2.8.2 Fuzzy Models

By the end of the 1990s, fuzzy modelling became popular owing to its ability in representing uncertainty and classifying fuzzy data.

Ismail and Gad [88] explored the potential of fuzzy logic in the field of Arabic signature verification. They employed fuzzy approaches in both the feature extraction process and the learning/classifying process. In their research, a set of endpoints and intersection points, which were regarded as critical points, were located and the distances to those of a referenced model were measured. The distances obtained were classified to be either Match, Near, Mid, or Far using a fuzzy rule set. After that, the Grade of a signature is determined from the number of members in each distance class. This grade or score was used to challenge another set of fuzzy rules to judge the class of a questioned signature. Despite the small signature corpus of 330 specimens for both training and testing, this work inspired further use of fuzzy logic in signature verification.

In their research, Hanmandlu [75] and Madasu [138] modelled the genuine signatures using the Takagi-Sugeno fuzzy model. In this approach, a signature is pre-processed to obtain its skeleton representation. This skeleton representation is divided horizontally into 8 equal portions. Each portion is further divided into 12 equal boxes (4 × 3). From each box, the summation of the angles of all foreground points with respect to the bottom left corner of the box is calculated. This value is normalized with the number of pixels in the box to obtain a feature prior to feeding into the fuzzy model. The authors innovatively introduced the shifted margins for the maxima and minima obtained from the membership functions. With the evidence of these margins, the model can absorb interpersonal variations whilst rejecting forgeries effectively. The experiments were conducted using a proprietary database with 40 writers with 1200 signatures (600 genuine and 600 forged). It is worth mentioning that this database has 200 skilled forgeries produced by expert forgers. These authors reported an extremely high accuracy of 99.8%.

The fuzzy-snake model has also been employed by Velez et al. to identify random forgeries [216].
2.8.3 Machine Learning Approaches

Statistical learning approaches have been employed in different research areas. In off-line signature verification, the frequently used statistical classifiers are neural networks, Support Vector Machines, and Hidden Markov Models. The following sub-sections review the investigation of these classifiers in off-line signature verification.

2.8.3.1 Neural Networks

Early notable investigations in the field of off-line signature verification using artificial neural networks includes the research of Mighell et al. [143]. In that work, the authors employed the back-propagation neural network for the construction of signature models. From the results obtained, the researchers concluded that the training process would require forgeries, otherwise all the signatures would be recognized as genuine in the test phase.

In another research, Qi and Hunt [179] compared the performance of neural networks against vector quantization (VQ) classifiers. Despite their incapability to represent non-linear class boundaries, VQ classifiers performed comparably to the neural networks, these researchers reported. Their explanation was that the decision hyper planes could not be established efficiently without simulated forgeries in the training phase.

Huang and Yan [83] employed two stage Multi-layer Perceptron (MLP) neural networks to examine the questioned signature image at different resolutions. The first stage includes several three layer perceptron networks. Each of which is dedicated to features extracted at a particular resolution. The combination stage employs the outputs of the networks in the first stage as its input to calculate the overall confidence. These researchers reported an accuracy as high as 90% using a proprietary database of over 3000 signature images.

Santos et al. [194] employed MLP-NNs with 640 inputs and 2 outputs to learn about the Euclidean distances between grid-based feature vectors extracted from all pairs of genuine signatures used for training. The number of hidden units was varied from 4 to 16. In this approach, 720 genuine signatures were employed in the training process. Given a questioned signature, its distance to reference genuine signatures was evaluated. The decision of genuineness for each distance is determined using the trained MLP. In the final step, the majority-voting scheme decides the authenticity of the questioned signature based on this information. The researchers reported FARs of 4.41%, 1.67%, and 15.67% for random, simple, and simulated forgeries respectively. The FRR was 10.33%. The overall error rate was 8.02%.

In another work, Armand et al. [11] compared the performance of the Resilient Back Propaga-
tion (RBP) and Radial Basis Function (RBF) neural networks in conjunction with the enhanced Modified Direction Features. The RBF-NN, with the accuracy of 91.21%, reportedly outperformed the RBP-NN. In their work, the number of genuine signatures and forgeries employed for training was 18 and 22 respectively.

Generally speaking, MLPs have no mechanism to minimize the risk of misclassification [96]. Unlike other binary classification problems, the class of targeted and skilled forgeries is unknown in signature verification. To overcome this obstacle, some researchers in the field proposed the adoption of pseudo-forgeries [83], which are generated using genuine signatures.

2.8.3.2 Hidden Markov Models

A Hidden Markov model (HMM) [184] is a statistical Markov model in which the system being modelled is assumed to be a Markov process with unobserved states. It is defined by a set of states and state-changing probability rules $\lambda = \{A, B, \pi\}$, where $A$ is the state transition probability distribution, $B$ is the observation symbol probability distribution in states, and $\pi$ is the initial states distribution.

The choice of an appropriate topology for HMMs is crucial as the topology is strongly related to the nature of the feature extraction technique employed. Whenever the feature extraction technique is grid-based, researchers often adopted the left-right topology [53]. Researchers in the field of (Western) signature verification often justify this choice by the fact that the direction of progress of signatures are often from left to right. If the feature extraction technique happened to utilise polar coordinates, the ring topology can be employed.

Apart from the choice of topology, probability density functions are another important aspect in the construction of HMMs. Among the functions, many researchers prefer discrete density [53, 99]. The parameters of an HMM is computed from a set of sequences of states using a forward-backward algorithm such as the Baum-Welch algorithm [184]. Given an HMM, the probability that it produces a sequence of observations can be computed efficiently using dynamic programming.

![Figure 2.8.2: The flow chart of a typical discrete HMMs signature verification system](image)

Many researchers regard signing as a Markov process and investigate the application of HMMs
for the signature verification process. Usually, an HMM-based automatic signature verification system consists of many HMMs, each model for a writer’s signature [99]. A general system is depicted in Figure 2.8.2. The authenticity of a questioned signature is determined by comparing the probabilities produced by the HMMs.

### 2.8.3.3 Support Vector Machines

SVM is a statistical learning technique developed by Vapnik [212] and first introduced in 1998. These classifiers implement the idea of mapping the input vectors $\mathcal{X}$ into a high-dimensional feature space $\mathcal{Z}$ through a non-linear mapping. In such a new feature space, the optimal separating hyperplane is constructed using a subset of training samples called the support vectors. It is also based on a structural risk minimization principle (SRM). Two main objectives of the SRM induction principle are to control the empirical risk on the training samples and to control the capacity of the decision functions used to obtain that risk value.

![Figure 2.8.3: Feature mapping](image)

A decision function of an SVM has the form of:

$$ f(x) = \text{sign}(w \cdot x + b) $$

(2.8.1)

Given a set of training vectors $S$ with $l$ pairs $(x_i, y_i)$ of samples:

$$ S_l = \{(x_1, y_1), ..., (x_l, y_l)\} \in \mathcal{R}^n, y_i \in \{-1, +1\} $$

(2.8.2)

Each of these samples belongs to either two classes $W_1(y_i = +1)$ or $W_2(y_i = -1)$.

The SVM finds the hyperplane with the maximum Euclidean distance from the training set. According to the SRM principle, there will be only one optimal hyperplane (Figure 2.8.4a) with the maximal margin $\delta$ defined as the distances from the hyperplane to the closest points of the two classes.
Dealing with non-separable training sets as illustrated in Figure 2.8.4b, the $i^{th}$ misclassified sample is assigned a slack variable, representing the magnitude of the classification error. The SVM solution for a non-separable training set can be found by keeping the upper bound on the Vapnik-Chervonenkis dimension minimized [31] and by minimizing an upper bound on the empirical risk with the following minimization:

$$\text{Minimize } \langle w \cdot w \rangle + C \sum_{i=1}^{l} \xi_i$$

Subject to

$$y_i(\langle w \cdot x_i \rangle + b) \geq 1 - \xi_i \forall i = 1, ..., l$$

(2.8.3)

where $C$ is the regularization constant determining the trade-off between the empirical error and the complexity term. The parameter is chosen by the user. A larger value of $C$ corresponds to higher penalty errors.

2.7.5.1 Kernel

It is a fact that mapping learning vectors into another feature space can greatly simplify the classification problem. This has been known for a long time within the machine learning area.

Minsky and Papert’s example [134] showed how a single layer perceptron failed to learn XOR logic. On the left, a single line cannot separate the black dots from the white dots. With additional dimension $z = x \times y$, this can be done as shown in the right figure.
logic in a 2-D space. When mapping the XOR logic to a 3-D space, the problem could easily be solved as depicted in Figure 2.8.5.

A kernel is a function $K$, such that for all $x, y \in X$:

$$K(x, y) = \langle \phi(x) \cdot \phi(y) \rangle$$

where $\phi$ is a mapping from feature space $X$ to another feature space $F$.

Popular kernels employed by researchers are:

- **Linear kernel**:

$$K(x, y) = (x \cdot y)$$  \hspace{1cm} (2.8.4)

- **Polynomial kernel**:

$$K(x, y) = (ax \cdot y + b)^d$$  \hspace{1cm} (2.8.5)

- **RBF kernel**:

$$K(x, y) = e^{-\frac{\|x-y\|^2}{\sigma^2}}$$  \hspace{1cm} (2.8.6)

Justino et al. [98] compared the performance of SVMs and HMM classifiers under two specific conditions. The first one is the number of genuine samples used for training and the other one is the use of different types of forgeries. A relatively large database of 100 signature sets with 30 genuine signatures for each set was employed in this research. Preliminary results from this research indicated that the results produced by the linear kernel were much better than the polynomial kernel. Under both conditions, SVMs showed better results than HMMs especially in recognizing forgeries. When there are more genuine signatures presented to the classifiers in the training process, the FRR curve for SVMs fell exponentially while the FAR increased linearly and mildly. The FRR and FAR (for targeted forgeries) were 3% and 4% respectively when the classifiers were trained using 20 genuine samples. The authors indicated that SVMs were well adapted to absorb intra-personal variability and allowed reasonable imitations. However, the signature database employed in this research was not open to the research community and few details are known about that signature database, such as how it is constructed, the number of simple forgeries, and the number of simulated forgeries.

In other research, Ferrer et al. [57] compared the performance of SVMs, HMMs, and Euclidean distance classifiers over a specific set of features. They obtained results indicating that SVMs performed slightly worse than HMMs and that the polynomial kernel performed better than the linear kernel, which is different to the findings of Justino et al. [98]. When the RBF kernel was in use, the FRR and the FAR for targeted forgeries were as low as 15.41% and 13.12% respectively (using targeted forgeries for training). It is worth mentioning that the database employed in this research is an open database.

In a more recent work, Friaz-Martinez et al. [62] compared and reported that SVMs outperformed MLPs. SVMs also reportedly outperformed the RBP and RBF neural networks in Nguyen et al.’s investigation employing the enhanced Modified Direction Feature [153].
2.8.4 Distribution of Inter-class and Intra-class Distances

In this approach, the distributions of inter-class and intra-class distances are assumed to be of some known statistical distribution. To determine the authenticity of a questioned specimen, its “distances” to the known genuine specimens are first established. The authenticity of the questioned signature is decided by comparing the likelihood that the obtained distances belong to the genuine-genuine distribution with the forgery-genuine counterpart [204, 202, 177]. In the literature, it is often assumed that the genuine to genuine distances are Gaussian distributed variables whilst the forgery to genuine distances follow Gaussian or Gamma distributions. An advantage of employing the distances is that from a small population of \( n \) specimens, a large number of \( \binom{n}{2} \) distances can be computed to facilitate the calculation of the parameters.

In [178], the authors investigated the performance of the GCS features [200] in conjunction with a Bayesian classifier using this approach. Both the genuine to genuine and the forgery to genuine distances were assumed to be of Gaussian distributions whilst the controlling parameters \( \sigma^2 \) and \( \gamma \) are of Gaussian and Gamma distributions respectively. Using the Bayesian rule, the authors established the relationship between the distribution of the distances of a questioned signature to the genuine specimens and the distribution of the:

\[
N(\theta_g | \mu_g, \tau^2_g) \propto N(X_g | \theta_g, \sigma^2_g) \times N(\theta_g | \mu_{g0}, \tau^2_{g0}) \tag{2.8.7}
\]

\[
N(\theta_f | \mu_f, \tau^2_f) \propto N(X_f | \theta_f, \sigma^2_f) \times N(\theta_f | \mu_{f0}, \tau^2_{f0}) \tag{2.8.8}
\]

The mean hyper-parameters \( \mu_{g0}, \mu_{f0} \), and variance hyper-parameters \( \tau^2_{g0}, \tau^2_{f0} \) were determined using a set of known genuine signatures and their forgeries. The probabilities that a given distance belongs to the genuine/forgery distributions are derived from the above equations as follows:

\[
P(t_i | G) = \int_{-\infty}^{\infty} P(t_i | \theta_g, \sigma^2_g) \times P(\theta_g) \times P(\sigma^2_g) d\theta_g d\sigma^2_g \tag{2.8.9}
\]

\[
P(t_i | F) = \int_{-\infty}^{\infty} P(t_i | \theta_f, \sigma^2_f) \times P(\theta_f) \times P(\sigma^2_f) d\theta_f d\sigma^2_f \tag{2.8.10}
\]

By Laplace Approximation, these integrals become:

\[
P(t_i | G) = \frac{A_g}{2\pi} e^{-\frac{A_g(t_i - \mu_g)^2}{\tau^2_g}} \tag{2.8.11}
\]

\[
P(t_i | F) = \frac{A_f}{2\pi} e^{-\frac{A_f(t_i - \mu_f)^2}{\tau^2_f}} \tag{2.8.12}
\]

where \( A_g = \frac{n+5}{nk_g\sigma^2_g} \); \( A_g = \frac{m+5}{nk_f\sigma^2_f} \); \( k_g = 1 + \frac{\tau^2_g}{\sigma^2_g} \); and \( k_f = 1 + \frac{\tau^2_f}{\sigma^2_f} \). \( m \) and \( n \) respectively are the number of genuine-genuine and forgery-genuine distances employed to estimate the hyper-parameters. The log-likelihood for a given signature is:

\[
LLR = \log\left( \prod_{i=1}^{n} \frac{P(t_i | G)}{P(t_i | F)} \right) \tag{2.8.13}
\]

A signature is classified genuine if its \( LLR \geq 0 \), otherwise it is a forgery.

Using the template matching approach, Yoshimura et al. (1995) reported an EER of 19.2% for
Japanese signatures [226]. One drawback of Yoshimura’s approach is the usage of stroke width, which may vary depending on the pen used to sign, as a match criterion. Taking this problem into account, Ueda normalized the strokes by blurring the one pixel Hildich skeleton of the signature using a fixed point-spread function [211]. Eventually, this improvement obtained significantly better results with an EER of 9.1%.

Considering the assumption that the variations of features and strokes of human signatures are limited to a finite range, Fang et al. investigated the approach of building the statistics of such variations from the training set [55]. There is no doubt that with more samples used in the training process, the higher the accuracy of the statistics. In their research, Fang et al. adopted the leave-one-out method, which employed up to 23 genuine signatures in the training process, and obtained an EER of 22.3%. A similar approach has been conducted by Jing et al. [94] using features extracted from Gabor transformed images. The authenticity of a questioned signature is determined by looking at its Mahalanobis distance, as compared to other known specimens. A remarkably low EER of 9.9% was reported despite the disadvantage of using a relative large number of genuine specimens for training. It should be noted that an automatic signature verification system would be less convenient if it requires more genuine specimens in order to construct writer profiles.

### 2.8.5 Combination of Multiple Expert Decisions

Majority-voting or multi-expert decisions are often employed when the importance of experts’ opinions are relatively equal.

Other researchers such as Dimauro [40], Allgrove [6] et al. suggested that by combining the outputs of different classification schemes, a system could obtain a higher accuracy. In this way, the majority voting scheme also exposes the system to the risk of the ‘tyranny of majority’. This is the case where the minority is overwhelmed by the majority’s incorrect decision. In their research, Allgrove and Fairhurst suggested that signature identification solutions generally do utilize all sources of information effectively and a combination of different approaches may increase overall accuracy. To test the idea, the authors used a majority voting scheme to combine the results of different approaches on a database of 6027 specimens from 200 subjects. Some combinations were reported as providing better identification rates whilst others worse. However, information about the experimental configuration was not detailed enough for validation purposes. Due to insignificant differences between the voting results and the decisions from each expert, the combination of results through majority voting needs further investigation.

Another technique to combine expert decisions is to employ a statistical learning machine. This approach makes the result combination process more ‘intelligent’ than the pure majority voting scheme. In their research, Huang and Yan [83] employed a two stages network to assess the questioned signature image of different resolutions. The first stage includes several three-layer perceptron networks. Each of them is dedicated to a resolution at which the features were extracted. The combination stage uses the outputs of the first stage networks as its inputs to calculate the overall confidence. The researchers reported an accuracy as high as 90% using a proprietary database of over 3000 signature images.

Similarly, Baltzakis [14] employed an RBF neural network to conclude the decisions returned by three neural networks and one Euclidean distance classifier for the random forgeries detection problem. In their proposed verification scheme, each neural network was responsible for a group
of features extracted: global, grid (pixel densities), and texture features (co-occurrence matrices). The database employed in their research consisted of signatures from 115 individuals, with approximately 15 to 20 genuine signatures per writer. The researchers reported an average FRR and FAR of 3% and 9.8% respectively.

In [194], Santos et al. employed majority voting to combine the results of the authenticity assessments of a signature being questioned. In this work, the researchers first trained a neural network using the genuine-genuine and genuine-random forgery distances. Given a questioned signature, its distances to the reference genuine specimens were computed. The neural network, trained previously, is then employed to determine whether each distance is either of the genuine-genuine class or the genuine-random forgery class. The final decision is made by majority voting. These researchers reported an AER of 13% in conjunction with an FAR of 4.41% and 1.67% for random forgeries and simple forgeries respectively. Figure 2.8.6 illustrates this approach.

### 2.9 Performance Evaluation

The ultimate target of verification systems is to be capable of accepting every genuine signature produced in normal conditions and rejecting forgeries of all types. However, in reality, verification systems often produce erroneous verdicts. A signature verification system may reject genuine signatures or accept forgeries at the rates respectively called *false rejection rate* (FRR) and *false acceptance rate* (FAR). In the literature, these error rates can also be called *Type I* and *Type II, or false negative rate* (FNR) and *false positive rate* (FPR). These rates are all measured in percentages. In signature verification research, the accuracies of verification systems are often reported using these rates. Usually, any attempt to decrease the one rate would cause the other to increase and vice versa. As these rates often vary disproportionately, the performance of a verification system is usually reported at the point where the average value of these error rates

![Diagram](image-url)
(AER) is at its minimum [174]. It should be noted that the definition of the AER varies in the
literature. Although the majority of researchers define the average value of the FRR and the FAR
for simulated forgeries (FAR2) as:

$$\text{AER} = \frac{\text{FRR} + \text{FAR2}}{2} \tag{2.9.1}$$

some compute the AER using the FAR values for every type of forgery. In their research, Batista
et al. [16, 17] employed the formula:

$$\text{AER}(\gamma) = \frac{\text{FNR}(\gamma) + \text{FPR}(\gamma)_{\text{rand}} + \text{FPR}(\gamma)_{\text{simp}} + \text{FPR}(\gamma)_{\text{skil}}}{4} \tag{2.9.2}$$

Apparently, an AER produced by Equation 2.9.2 would be significantly smaller than Equation
2.9.1 as the error rates FPR(\gamma)_{\text{rand}} and FPR(\gamma)_{\text{simp}} are usually much lower than the FNR(\gamma) and FPR(\gamma)_{\text{skil}}.

The equal error rate (EER) is another commonly used measurement employed by researchers
in signature verification. This is the rate of the FRR and the FAR of the most challenging type
of signature being investigated when these rates are equal. Usually, the EER tends to be slightly
higher than the AER. In [152], the researchers suggested that when the signature corpus employed
is representative, the lowest AER approximates the EER (See Figure 2.9.1).

Researchers in signature verification also evaluate and compare performance of off-line signature
verification systems using the receiver operating characteristic (ROC) curve [35, 157, 18]. The ROC
curve is created by plotting the FAR as a function of the FRR. The area under the curve (AUC)
is equivalent to the probability that the classifier will rank a randomly chosen positive sample
higher than a randomly chosen negative sample [56]. The closer to 1 the AUC is, the better the
performance.
Chapter 3

RESEARCH METHODOLOGY AND PROPOSED TECHNIQUES

From the literature reviewed in Chapter 2, it can be perceived that off-line signature verification is an active research area. The challenging nature of the problem has attracted the attention of many researchers. A great number of approaches as well as feature extraction techniques have been proposed and investigated. Despite numerous promising results reported, the unavailability of a successful commercial off-line signature verification system is clear evidence that there still exists a significant gap between research and reality. There are several issues with the methodologies in the literature that contributed to this situation. Many of those issues were mentioned and discussed in detail in Chapter 2. The present chapter is devoted to the methodology employed in this research and the proposed techniques in an attempt to overcome the obstacles existing towards the construction of an effective automatic signature verification system.

Firstly, Section 3.1 describes in detail the research methodology employed. After that in Section 3.2, the proposed contour-based intersection analysis framework will be discussed. Finally, Section 3.3 details the feature extraction techniques that were investigated in the present research. The newly proposed features described in the last section include Gaussian Grid, Curvature Map, Variance, New Ratio, Energy, Trajectory Length, and Moment-based features. Amongst the features, Curvature Map is a novel local feature that utilises the recovered trajectory.

3.1 Research Methodology

The Research Methodology section begins with the proposed off-line signature verification system in Section 3.1.1. Section 3.1.2 follows by justifying the choice of GPDS corpus as the research database. The characteristics of the selected database will also be described in this section. After that, Section 3.1.3 justifies why the SVMs have been chosen as the classifiers in the current research. Finally, Section 3.1.5 discusses how the performance of the proposed system was evaluated.
3.1.1 Proposed System

In the proposed off-line signature verification system, the profile of each user of the signature verification system is established using genuine signatures and random forgeries which are readily available. A two-class classifier is employed to create the user profile using these signatures. When being given a questioned signature, the classifier refers to the established profile to decide whether it is a genuine or a forgery. A system such as this is classified as a writer-dependent system [200].

Firstly and most importantly, it is assumed that the parameters of the verification system are universal and can be used for any additional user. This guarantees the scalability of the proposed system. Secondly, it is assumed that these parameters can be found in the parameter tuning phase using a large and representative database and can be reliably used in the deployment phase. In the current research, the target function, when determining these parameters, is to minimize the average error rate (AER) for genuine signatures and simulated forgeries. Figure 3.1.1 illustrates the main processes of the proposed signature verification system.

3.1.2 Off-line Signature Database

As previously mentioned in Section 2.4, the reliability of research in signature verification is highly influenced by the signature database employed. A research database must include a large number of signature sets which have been carefully collected in order to represent the population. It should also be accessible by the research community to facilitate less bias comparisons and validation of different approaches. Amongst the publicly available signature databases, the MCYT [58] and GPDS [57, 213] are the most frequently employed databases in off-line signature verification research.

Each signature set of the GPDS database consists of 24 genuine signatures and 30 simulated forgeries. This is significantly larger compared to the publicly available version of the MCYT corpus with 15 genuine signatures and 15 simulated forgeries. For each authentic signer, all the genuine signatures were collected in a single day’s writing session. The authors were asked to produce their signatures on a sheet of paper each consisted of 24 boxed signing areas. The signatures were requested to be within the boxes. The dimensions of boxes are those commonly found in documents, forms, or cheque.

To create the simulated forgeries, each lay forger was exposed to a genuine signature for each of
5 randomly selected individuals. They were asked to produce 3 forgeries for each genuine signature in a single day’s writing session. The genuine signatures of an individual shown to the forgers were chosen randomly from his/her 24 genuine signatures. Therefore, there are 30 skilled forgeries made by 10 forgers for each genuine signature using 10 different genuine specimens. The forgers were allowed to practice their forgeries as long as they wished using the static image of genuine specimens provided before producing the forgeries. This is to reflect the fact that forgers usually have plenty of time to practise forging the targeted signature.

The GPDS signature database has been released to the research community in increasing quality as well as number of signature sets during its development. In the current release, there are 960 sets of signature at the resolution of 300dpi. The URL for downloading this database is: http://www.gpds.ulpgc.es/download/index.htm Compared to other publicly available signature databases, the GPDS corpus has the largest number of signature sets as well as the highest number of genuine and forgeries per signature set. With these considerations, it was decided to employ the GPDS as the research database. Among various versions of the GPDS, the GPDS-160 database has been more commonly used by researchers. In order to obtain comparable results, this version was employed in the present research.

3.1.3 Classification Method

The literature reviewed in Section 2.8 has shown that many classification techniques have been investigated for automated signature verification. Researchers generally agree that the classifiers contribute significantly to the overall performance of systems. In order to obtain better results, the choice of which classifier must take into account the characteristics of the feature extraction technique employed as well as the problem.

As previously mentioned in Section 1.2, the off-line signature verification problem is different from many other two-class separation problems. Firstly, only the genuine class is known. Secondly, this known class can only be characterized using a very limited number of specimens. Therefore the classifier must have high generalization power. In other words, it should be capable of finding an optimal separation hyper-plane that minimizes the risk of misclassification.

In off-line signature verification, the performance of the Gaussian kernel SVMs was favourably compared against RBF and RBP neural networks [153]. Researchers explained the outcomes by the ability of the SVMs in maximizing the margin from the classes to the separation hyper-plane which is unavailable in the RBF and RBP neural networks. Other researchers such as Justino et al. [98] reported that SVMs produced better results compared to HMMs despite HMMs having been employed intensively in the area. The employment of HMMs in signature verification also has an issue which is that they are not proficient in properly applying features extracted from static images by Markov processes. In fact, there has never been any research investigating HMMs about this issue. Lately, support vector machines reportedly outperformed the Squared Mahalanobis Distance classifier [154]. All the above comparative advantages of SVMs strongly encouraged their adoption as the classification technique for this research.

There are a number of aspects, which need to be considered in the experiments using SVMs so that better results can be achieved. Firstly, the training of the SVMs requires samples from both the positive and the negative classes like any other binary classifier. The unavailability of simulated forgeries for training in the real world suggests that the learning process can only employ random forgeries for the negative class. If the optimal set of parameters were determined with the
Table 3.1: Experimental Settings

<table>
<thead>
<tr>
<th>Phase</th>
<th>Genuine</th>
<th>Forgery</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>Random</td>
</tr>
<tr>
<td>Training</td>
<td>12</td>
<td>400</td>
</tr>
<tr>
<td>Testing</td>
<td>12</td>
<td>59</td>
</tr>
</tbody>
</table>

involvement of the simulated forgeries, it can’t be used in deployment since the enrolment of any additional individual would require his own simulated forgeries, which is impractical. Secondly, an appropriate kernel function must be chosen for the SVMs, and in this case experimentation was conducted using the RBF kernel with different values of sigma. The trade-off between training error and margin was set to be 1000.

In the cursive handwritten character recognition experiments, the RBP and RBF neural networks were employed in addition to the use of SVMs. Preliminary experiments indicated that the training and classifying processes of these classifiers are significantly faster than the SVMs for a multi-class classification problem like cursive character recognition. The number of hidden units for RBP neural networks varied from 2 to 512 whilst the number of iterations was fixed at 10000. In the experiments with the RBF neural networks, the number of centres was changed exponentially from 128 to 16384.

3.1.4 Experimental Settings

In the experiments of this research, the classifiers were trained using genuine signatures and random forgeries. The simulated forgeries are only used in the testing phase in conjunction with random forgeries and genuine signatures, i.e. those that were not used for training. As the amount of training samples influence system accuracy significantly, it is necessary to employ an appropriate sample configuration that balances the requirements of a signature verification system.

On the one hand, the number of genuine signatures used for training should be as small as possible to increase the convenience and the practicality of the verification system. On the other hand, it needs to be large enough for the system to be reliable. A larger number of genuine signatures allow the user profile to be constructed with greater accuracy. Considering the above restrictions as well as the figures recommended by other state-of-the-art approaches [98, 57, 229], it was decided that 12 genuine signatures would be used for training in each experiment with each signature set. The remaining 12 genuine signatures were used for testing.

Although the random forgeries are readily available in actual scenarios, only a moderate number of random forgeries were employed for training to limit the training time. In each experiment with a signature set, 400 genuine signatures were taken from the other 100 signature sets, four signatures from each, as random forgeries for training. One genuine signature was also taken from every 59 remaining sets as a random forgery for testing. These settings are detailed in Table 3.1.

In order to increase the reliability and to avoid bias of the experimental results, each experiment was repeated 30 times for each signature set. Each time, the experiment employed a random set of training and testing samples. The random sample generation process used in this research guarantees that each of the 24 genuine signatures is employed, either for training or testing, in exactly \( \frac{30 \times 12}{24} = 15 \) sub-experiments and each of the 30 forgeries is employed for testing in exactly \( \frac{30 \times 15}{30} = 15 \) sub-experiments.
3.1.5 Evaluation of Verification Accuracy

The input for automatic off-line signature verification is both genuine and forged signatures. An ideal verification system should be capable of rejecting all the forgeries whilst accepting every signature produced by authentic writers under normal conditions. In reality, signature verification systems may wrongly reject genuine signatures as well as accept forgeries.

The rate at which genuine signatures are rejected and forgeries are accepted are called the False Rejection Rate (FRR) and False Acceptance Rate (FAR), respectively. Any attempt to decrease the one rate by changing the decision threshold is likely to increase the other and vice versa. In signature verification, researchers mainly focus on the problem of verifying genuine signature against simulated/skilled forgeries. Therefore, the performance of many signature verification systems in the literature are often reported by the lowest average value of the FRR and FAR for simulated forgeries (FAR2). This value is called the average error rate (AER) and is computed as follows:

\[
AER = \frac{FRR + FAR2}{2}
\]  

The above formula is also used to report signature verification accuracy throughout the present research.

The identification of random forgeries is considered a much easier problem. Many researchers have reported FAR1 rates below 0.1%. In the present research, the FAR1 rates are reported only for information purposes.

3.2 Intersection Analysis

From the literature reviewed in Section 2.7, it is observed that many trajectory-based features improve the performance of off-line handwriting recognition systems. A large proportion of the techniques reviewed require preprocessing or for the input patterns to satisfy one or more conditions. Undoubtedly, such constraints prevent these techniques from being more popularly deployed in off-line handwriting recognition systems. Table 3.2 presents the performance of significant trajectory recovery techniques along with their key experimental settings.

Employing a generic handwriting corpus, Plamondon and Privitera [175] obtained a success rate of 89%. Better results were reported when one or more constraints, such as crossing complexity, number of strokes or line width, were in place. Kato and Yasuhara [107] reported a recovery rate of 91.6% whilst Qiao et al. [181] reported the best recovery rate of 96%. In character recognition, Rousseau et al. [188] achieved a recovery rate of 93.3% when characters with unexpected models were removed from the experiments. Zou and Yan [234] reported a remarkably high recovery rate of 97.6% for numerals. In loop recovery, Doermann et al. [44] reported the best rate of 84%. The trajectory recovery framework employed by researchers often consists of three major processes: (i) Localise and analyse intersections. (ii) Identifying start points and end points. (iii) Search through the possible configuration space for the most likely trajectory.

Although there is diversity in the approaches, stroke continuity estimation is performed in many, if not all cases. It is used to detect and analyse ambiguous zones [175], to detect double traced lines, loops, and blobs [205]. However, most techniques found in the literature to date often estimate this value roughly. It is strongly believed that more precise estimation would significantly improve the recovery rate. Potential continuity functions could be derived from handwriting movement models.
<table>
<thead>
<tr>
<th>Author</th>
<th>Year</th>
<th>Material</th>
<th>Subject</th>
<th>Experimental Settings</th>
<th>Accuracy</th>
</tr>
</thead>
<tbody>
<tr>
<td>Boccignone et al. [27]</td>
<td>1993</td>
<td>Skeleton</td>
<td>Characters</td>
<td>10,000 characters by 20 individuals</td>
<td>97.0%</td>
</tr>
<tr>
<td>Abuhaiba et al. [1]</td>
<td>1995</td>
<td>Contour</td>
<td>Loops</td>
<td>2 writers, 65 strokes, 159 blobs</td>
<td>83.6%</td>
</tr>
<tr>
<td>Allen &amp; Navarro [5]</td>
<td>1997</td>
<td>Skeleton</td>
<td>Characters</td>
<td>250 dpi, 1248 characters by 12 individuals</td>
<td>91.6%</td>
</tr>
<tr>
<td>Zou and Yan [234]</td>
<td>1999</td>
<td>Skeleton</td>
<td>Numerals</td>
<td>NIST database</td>
<td>97.6%</td>
</tr>
<tr>
<td>Plamondon &amp; Privitera [175]</td>
<td>1999</td>
<td>Contour</td>
<td>Handwriting</td>
<td>200 city names, 1390 ambiguous zones</td>
<td>89.0%</td>
</tr>
<tr>
<td>Lallican &amp; Viard-Gaudin [117]</td>
<td>1999</td>
<td>Contour</td>
<td>Characters</td>
<td>260 characters by 10 writers</td>
<td>90.0%</td>
</tr>
<tr>
<td>Lallican et al. [118]</td>
<td>2000</td>
<td>Contour</td>
<td>Handwriting</td>
<td>IRONOFF [218] corpus, 20898 training/10448 testing words by 700 individuals from</td>
<td>80.0%</td>
</tr>
<tr>
<td>L’Homer [129]</td>
<td>2000</td>
<td>Contour</td>
<td>Characters</td>
<td>520 characters from NIST database</td>
<td>90.0%</td>
</tr>
<tr>
<td>Kato &amp; Yasuhara [107]</td>
<td>2000</td>
<td>Skeleton</td>
<td>Handwriting</td>
<td>200 dpi, 100 single stroke drawing</td>
<td>91.6%</td>
</tr>
<tr>
<td>Doermann et al. [44]</td>
<td>2002</td>
<td>Contour</td>
<td>Loops</td>
<td>1270 words by 5 individuals</td>
<td>84.0%</td>
</tr>
<tr>
<td>El-Baati et al. [51]</td>
<td>2005</td>
<td>Contour</td>
<td>Handwriting</td>
<td>50 Arabic words by 2 individuals</td>
<td>92.0%</td>
</tr>
<tr>
<td>Lau et al. [123]</td>
<td>2005</td>
<td>Skeleton</td>
<td>Signatures</td>
<td>350 online training / 300 off-line testing</td>
<td>90.0%</td>
</tr>
<tr>
<td>Nel et al. [148]</td>
<td>2005</td>
<td>Skeleton</td>
<td>Signatures</td>
<td>Assisted with online references, 710 single-path signatures by 50 authors</td>
<td>91.5%</td>
</tr>
<tr>
<td>Rousseau et al. [187]</td>
<td>2005</td>
<td>Skeleton</td>
<td>Characters</td>
<td>5800 characters, single-stroked, no hidden ends</td>
<td>87.0%</td>
</tr>
<tr>
<td>Nefedov [147]</td>
<td>2006</td>
<td>Skeleton</td>
<td>Characters</td>
<td>50 characters whose skeletons have up to 3 junction points</td>
<td>97.4%</td>
</tr>
<tr>
<td>Niels &amp; Vuupijl [156]</td>
<td>2006</td>
<td>Skeleton</td>
<td>Characters</td>
<td>UNIPEN corpus [71], 1 pixel line width, 3370 samples, samples contain skeleton artefacts are removed</td>
<td>88.0%</td>
</tr>
<tr>
<td>Qiao &amp; Yasuhara [183]</td>
<td>2006</td>
<td>Skeleton</td>
<td>Handwriting</td>
<td>UNIPEN corpus, 3 pixel line width, words contain 3 and/or 4 branches intersections only</td>
<td>93.7%</td>
</tr>
<tr>
<td>Qiao et al. [181]</td>
<td>2006</td>
<td>Skeleton</td>
<td>Handwriting</td>
<td>UNIPEN corpus, 3 pixel line width, 708,881 simulated static images 187 single-stroked offline handwriting images</td>
<td>96.0%</td>
</tr>
<tr>
<td>Rousseau et al. [188]</td>
<td>2006</td>
<td>Skeleton</td>
<td>Characters</td>
<td>5556 training/ 1852 testing multi-stroked lower case letters with unexpected models noisy image removed</td>
<td>83.3%</td>
</tr>
<tr>
<td>Steinherz et al. [205]</td>
<td>2009</td>
<td>Contour</td>
<td>Loops</td>
<td>540 loops from IRONOFF database</td>
<td>80.2%</td>
</tr>
</tbody>
</table>
such as Log-normal [69] or $\beta$-elliptic [20] approaches.

Before proceeding to the other aspects of the methodology, it is necessary to consider both the advantages and disadvantages of using each type of input information that can be used. As previously mentioned in Section 2.7.1, two types of input information for trajectory recovery are the skeletons and the contours. In the skeleton-based approach, the problem is often reformulated to a standard graph traversal problem, either using Eulerian or Hamiltonian circles. The major obstacles to this approach are the artefacts produced during skeletonisation. Whilst some researchers carefully examine the topology of artefacts for trajectory clues [107, 181], others simply consider those as graph vertices which join skeleton branches. From a holistic perspective, both of these ways of handling artefacts are inappropriate. The skeletonisation process itself may have already caused degradation and even loss of information.

Compared to their skeleton-based counterparts, contour-based approaches appear to be capable of better utilising available information. Each contour segment must correspond to a section of the pen trajectory. Furthermore, the visible section on one side of the contour may also be employed to estimate or recover the corresponding but occluded sections as well as the true trajectory.

In Plamondon and Privitera’s research [175], the ambiguous zones were located by using a sweep window of limited size to scan through the handwriting image. A window is selected for further analysis if all of the foreground pixels inside are connected and the shape of the connected zone belongs to a simple predefined taxonomy of maximum curvature points. Since signatures are more sophisticated than ordinary handwriting samples, with many other topologies of ambiguous zones, this technique may be inappropriate for analysing signatures.

Algorithm 3.1 Intersection Analysis Framework

1: Determine contour curvature extrema
2: Break the contours into separate contour segments
3: Find candidate pairs of segments
4: Calculate the bridging score for each pair of segments
5: Compute the optimal matching configuration

Algorithm 3.1 summarises the steps of the proposed intersection analysis framework. In the first step, the local curvature extrema are localised from the contour of the input handwriting. These curvature extrema are then used to break the contours into separate contour segments in the second step. As a contour segment should only be followed by another which is also involved at the intersection, a set of rules is employed to eliminate pairs of segments that do not satisfy this requirement in the third step. A matching score is then assigned for each possible match. In the final step, the optimal matching configuration is computed using the Hungarian algorithm.

3.2.1 Boundary Extraction

Boundary Extraction is an important step in the proposed framework. The decision of whether to match two strokes joining at an intersection is made by carefully examining the stroke boundaries. In order to facilitate further processing, it is necessary that each boundary pixel is adjacent to exactly two other boundary pixels. A boundary extraction algorithm as described in [162] was employed.

Firstly, a raster scan is used to search for foreground pixels. Whenever a foreground pixel is found, the number of its foreground neighbours in the up, down, left, and right directions is counted. If there are exactly 4 foreground neighbours then the pixel being considered is marked as an interior
Algorithm 3.2 Boundary Extraction
1: for each foreground pixel \( p \) do
2: if \( p \) has four foreground neighbours in directions UP, DOWN, LEFT, RIGHT then
3: mark \( p \) as interior
4: else
5: mark \( p \) as exterior
6: end if
7: end for
8: done \( \leftarrow \) false
9: while not done do
10: done \( \leftarrow \) true
11: for each exterior pixel \( p \) do
12: if \( \# \{ \text{exterior pixel } q \text{ adjacent to } p \} \neq 2 \) then
13: paint \( p \) as background
14: mark interior neighbours of \( p \) as exterior
15: done \( \leftarrow \) false
16: end if
17: end for
18: end while

![Signature Image and Boundary](image)

Figure 3.2.1: A signature image and its boundary representation

pixel. Secondly, all the interior pixels are recast as background pixels. For the remaining exterior pixels, the number of adjacent exterior pixels are counted and they are recast as background pixels if they have more than 2 exterior neighbours. To be considered neighbours, two adjacent pixels on the boundary \( I(x_1, y_1) \) and \( I(x_2, y_2) \) must satisfy all of the following conditions:

1. If \( |x_1 - x_2| = 1 \) and \( |y_1 - y_2| = 1 \) then \( I(x_1, y_2) = 0 \) and \( I(x_1, y_2) \neq 0 \) or \( I(x_1, y_2) \neq 0 \) and \( I(x_1, y_2) = 0 \)
2. If \( x_1 = x_2 \) and \( |y_1 - y_2| = 1 \) then \( I(x_i - 1, y_1) = \text{background} \) and \( I(x_i + 1, y_1) \neq \text{background} \) or \( I(x_i - 1, y_1) \neq \text{background} \) and \( I(x_i + 1, y_1) = \text{background} \)
3. If \( |x_1 - x_2| = 1 \) and \( y_1 = y_2 \) then \( I(x_i, y_1 - 1) = \text{background} \) and \( I(x_i, y_1 + 1) \neq \text{background} \) or \( I(x_i, y_1 - 1) \neq \text{background} \) and \( I(x_i, y_1 + 1) = \text{background} \)

This process is repeated until no more change occurs. It guarantees that each boundary pixel obtained has exactly two boundary neighbour pixels. The whole process is summarized in Algorithm 3.2. Figure 3.2.1 illustrates a signature and its boundary.

3.2.2 Determining Contour Curvature Extrema and Isolating Contour Segments

The first step of the proposed trajectory recovery framework is to identify every point on the boundary at which two distinct line segments intersect. At first glance, this seems to be an easy problem which is similar to finding points with maximum curvature locally. However, this problem is unsolved and the difficulty is two-fold.
Firstly, the curvature of a curve cannot be precisely computed given its discrete digitized representation. It can easily be confirmed that during the digitisation process, many curves can be contracted to a small set of discrete grid points. As a consequence, the curvature cannot be precisely computed and this information can only be approximated using externalities such as nearby/local supporting points. Secondly, local is a relative term and it adheres to the scale and region being considered. Different scope of supporting points under consideration will produce different sets of results. A larger supporting area tends to leave out points of smaller curvature whilst a smaller supporting area will burden subsequent processes and will very much be affected by noise. Besides, the choice of a relatively small supporting area does not guarantee that every point of interest (POI) will be included.

**Algorithm 3.3** findInSegment

1. \( \text{result} \leftarrow \emptyset, \text{MaxSet} = \emptyset, C_{\text{max}} = 0 \)
2. repeat
3. for Point \( p \) within the range of radius from either ends of the input segment do
4. Assign \( \text{MAX} \) as the pseudo curvature \( C_p \) of \( p \)
5. end for
6. if every points has been assigned \( \text{MAX} \) then
7. break
8. end if
9. for each unassigned point \( p \) do
10. Keep track of the left and right assistant points \( (p_l, p_r) \) whose distances to \( p \) closest to \( \text{radius}_{\text{supportIdx}} \)
11. Assign altitude of point \( p \) w.r.t. the line segment \( p_l p_r \) as the pseudo-curvature of \( p \left( C_p \right) \)
12. if \( C_p > C_{\text{max}} \) then
13. \( C_{\text{max}} \leftarrow C_p \)
14. \( \text{MaxSet} \leftarrow \{p\} \)
15. else
16. if \( C_p = C_{\text{max}} \) then
17. \( \text{MaxSet} \leftarrow \{p\} \)
18. end if
19. end if
20. end for
21. for each \( \text{cand} \in \text{MaxSet} \) do
22. if supporting area of assistant points of \( \text{cand} \) is too small then
23. continue
24. end if
25. if \((C_{\text{cand}} - C_{\text{cand.left}} >= \text{THRESHOLD}) \) and \((C_{\text{cand}} - C_{\text{cand.right}} >= \text{THRESHOLD}) \) then
26. \( \text{result} \leftarrow \text{cand} \)
27. \( \text{result} \leftarrow \text{findInSegment}(0, \text{cand}, \text{supportIdx}) \)
28. \( \text{result} \leftarrow \text{findInSegment}(\text{cand}, , \text{supportIdx}) \)
29. return \( \text{result} \)
30. end if
31. end for
32. increase supportIdx
33. until \( \text{radius}_{\text{supportIdx}} > \text{max}_{\text{support}} \) or \( \text{radius}_{\text{supportIdx}} >= \text{RADIUS} \)
34. return \( \text{result} \)

In the present research, it is proposed that a flexible size supporting area should be employed. This enables the selection of high curvature maxima in smaller scale and low curvature maxima in larger scale. Starting from a relatively small size, which enables the location of extrema with high curvature, the size of the supporting area grows gradually. Each time an extreme point is found,
the contour is segmented and the process restarts with the new segments obtained. When no additional extreme point is found using the current supporting area, the size is increased. The size values of the supporting area are the distinct values of the set \( \{ \sqrt{i^2 + j^2} : i = 1\ldots T, \ j = 1\ldots T \} \) where \( T \) is a predefined threshold. The algorithm stops when the size of the supporting area exceeds another predefined threshold.

For each value of supporting area, the pseudo curvature is computed for each point on the contour being considered as follows. At each point, namely \( A \), a circle with the radius of the supporting area is drawn around the point. This circle cuts the contour at two distinct points \( B \) and \( C \). The pseudo curvature at \( A \) for the current supporting area is the altitude \( AH \) of triangle \( ABC \). Point \( A \) is considered an extreme point of curvature if the pseudo curvature of \( A \) is greater than those of \( B \) and \( C \) and the differences are larger than a certain threshold. The whole procedure is presented in Algorithm 3.3.

**3.2.3 Candidate Pairs of Segments**

After the curvature local extrema have been located, the extrema where the contour is convex are discarded. The remaining points are then used to break the continuous contours into separate segments. Given \( N \) segments, there would be \( C_N^2 \) possible pairs. However, only nearby segments should be considered. In order to reduce the workload for subsequent processes, a set of rules was implemented to discard pairs of contour segments that should not be bridged.

Two segments can be bridged only if all of the following conditions are satisfied:

1. A straight line or a curve consisting of foreground pixels exists between the last point of the preceding segment and the first point of the succeeding segment. If the stroke is continuous then this condition must be true.

2. Both segments are on the same side of the contour with regards to the direction of progress. This condition is illustrated in Figure 3.2.2. In that figure, segment \( AA' \) can be bridged with segment \( CC' \) but not with segment \( BB' \).

3. The bridging of the two non-continuous segments does not form a U-turn in the direction of progress.

**3.2.4 Bridging Score**

Prior to the computation of the best matching pairs of the contour segments, bridging scores are assigned to every pair of segments (of those that can be matched). The bridging score of every
two separate segments represents the likelihood that the one segment is followed by the other.

For each given pair of contour segments, the process begins with the nearby terminals of the two segments. Gradually, there is an effort to include more points from both segments and an attempt is made to fit these points to algebraic curves. Ideally, the algebraic curves should be constructed using handwriting models. However, the scope of this research is limited to straight lines and circles only. The minimax geometric distance fitting algorithms employed in this research are the straight line fitting algorithm and circle fitting algorithms. In each step, the algorithms check whether the maximum geometric distance from the points included in the curves exceed a predefined threshold:

\[
\text{max}(\text{dist}(p_i, \text{obj})) \leq \text{THRESHOLD}
\]

These fitting algorithms were executed in ascending order of complexity: line-fitting, circle-fitting.

**Algorithm 3.4** Compute bridging score

**Require:** Prev Segment

**Require:** Next Segment

1. \( \text{prev}_\text{from} \leftarrow \text{Prev}.\text{last} - 1 \)
2. \( \text{prev}_\text{to} \leftarrow \text{Prev}.\text{last} \)
3. \( \text{next}_\text{from} \leftarrow 0 \)
4. \( \text{next}_\text{to} \leftarrow 1 \)
5. \( L_{\text{prev}} \leftarrow \text{length from prev}_\text{from} \text{ to prev}_\text{to} \text{ along Prev} \)
6. \( L_{\text{next}} \leftarrow \text{length from next}_\text{from} \text{ to next}_\text{to} \text{ along Next} \)
7. \( \text{extendable} \leftarrow \text{true} \)
8. \( \text{obj} \leftarrow \text{null} \)
9. **while** extendable **do**
10. \( \text{extendable} \leftarrow \text{false} \)
11. **if** \( L_{\text{prev}} \leq L_{\text{next}} \) **then**
12. **if** \( \text{prev}_\text{from} > 0 \) **then**
13. \( \text{decrease prev}_\text{from} \)
14. \( \text{Update } L_{\text{prev}} \text{ w.r.t. new prev}_\text{from} \)
15. **if** \( \text{obj}=\text{fit}(\text{Prev, prev}_\text{from, prev}_\text{to, Next, next}_\text{from, next}_\text{to}) \neq \text{null} \) **then**
16. \( \text{extendable} \leftarrow \text{true} \)
17. **else**
18. \( \text{increase prev}_\text{from} \)
19. \( \text{restore } L_{\text{prev}} \)
20. **end if**
21. **end if**
22. **else**
23. **if** \( \text{next}_\text{to} < \text{Next.size} \) **then**
24. \( \text{increase next}_\text{to} \)
25. **update } L_{\text{next}} \text{ w.r.t. new next}_\text{to} \)
26. **if** \( \text{obj}=\text{fit}(\text{Prev, prev}_\text{from, prev}_\text{to, Next, next}_\text{from, next}_\text{to}) \neq \text{null} \) **then**
27. \( \text{extendable} \leftarrow \text{true} \)
28. **else**
29. \( \text{decrease next}_\text{to} \)
30. **restore } L_{\text{prev}} \)
31. **end if**
32. **end if**
33. **end if**
34. **end while**
35. **return** \( L_{\text{prev}} + L_{\text{next}} \)

As soon as no additional points can be added and fit to algebraic curves, the length of the
curve from the last point included as part of its segment bridging end, is returned as the bridging score. The whole process is detailed in Algorithm 3.4. In practice, contour ends may have been bent at the intersection due capillary forces. The degree of the bending depends on the speed of pen movement and the characteristic of the ink as well as writing surface. Although appropriate processing should be employed to properly correct and to compensate those bent ends, the present research simply discards some pixels of both ends gradually and chooses the highest score.

3.2.4.1 Line Fitting

The minimax geometric distance line fitting algorithm was executed as follows. Firstly, the Graham scan algorithm [195] was employed to find the smallest convex hull of the input point set. After that, for each edge $A_iA_{i+1}$ of the line $ax + by + c = 0$ of the convex hull found, the corresponding vertex $C(x_C, y_C)$ that has the maximum distance to the edge is identified. This distance is then compared to other distances found with other edges to find the pair (edge, vertex) that has the smallest distance:

$$\text{distance} = \min_{A_iA_{i+1}} \max_C \frac{|ax_C + by_C + c|}{\sqrt{a^2 + b^2}}$$  \hspace{1cm} (3.2.2)

The minimax geometric distance fitting line is the one located in the middle of the edge and vertex that is parallel to the edge.

The complexity of this algorithm is $O(n \log n)$ as the most time-consuming process is finding the convex hull which is of $O(n \log n)$.

3.2.4.2 Circle Fitting

Similar to the minimax geometric distance line fitting algorithm, the algorithm for circle fitting involves finding the convex hull [49]. Roy and Zhang [189] proposed an algorithm that supposedly solved the problem in $O(n^2)$ time. Their algorithm utilized the Voronoi diagrams of the inner and outer convex hull. As performance is not the top concern in this research, a more simple algorithm with $O(n^3)$ complexity was employed instead.

For each edge of the convex hull of the input point set, the algorithm considered the line segment connecting every pair of points. The intersection of the bisectors of the edge and the line segment is the centre of the candidate circle. The radius of the candidate fitting circle is one half the difference between the distances of the furthest point and nearest point to the centre. If the distance of the segments ends and the candidate fitting circle is smaller than the best distance found, the distance of the remaining points to the candidate fitting circle are computed. The candidate fitting circle replaces the best fitting circle found if those distances are all smaller than the maximum distance associated with the current best fitting circle.

3.2.5 Trajectory Estimation

After the bridging scores have been computed for every bridgeable pair of contour segments, the next step is to determine the optimal bridging configuration. Although one segment can be joined to more than one other segment, the true trajectory only allows one connection. It is important to find the most suitable preceding and succeeding segments for every contour segment. To solve this problem, the maximum weighted bipartite matching algorithm is employed in the proposed framework.
The weight matrix \((M)\) for the minimum weighted bipartite matching algorithm is created through the following. Each segment is assigned a distinct number. Whenever a bridge with score \(s_{ij}\) can be created between segment \(i\) and segment \(j\), the element \(M_{ij}\) is assigned with the value \(C_1 - s_{ij}\). Otherwise, \(M_{ij}\) is assigned with a very large value \(C_2\). \(C_1\) and \(C_2\) should be chosen so that \(C_1 - s_{ij} \geq 0\) and \(C_2 \gg C_1 - s_{ij}\) \(\forall i,j\). In the present research \(C_1\) is 10,000 and \(C_2\) is 20,000. The output of the algorithm is an arrangement that maximises the sum:

\[
\Sigma_{i=1}^{N} M_{ij} : j_i \in \{1..N\} \land j_i \neq j_i \forall i \neq i^*
\]  

(3.2.3)

Although the arrangement obtained may not resemble the true trajectory, it provides a good approximation.

### 3.3 Feature Extraction

The literature reviewed in Section 2.6 of Chapter 2 showed that whilst the advances in approaches and classifiers are slowly emerging, a large number of feature extraction techniques have been proposed and investigated. Some researchers suggested that an appropriate combination of both global and local features should be employed in order to construct robust signature verification systems [161, 11, 58]. Undoubtedly, feature extraction plays a crucial role in off-line signature verification. The major target of the present research is to propose high performance novel feature extraction techniques for the off-line signature verification problem following an in-depth analysis of the existing state-of-the-art techniques.

It is generally agreed that a good feature extraction technique must be capable of capturing the intrinsic characteristics of each class of patterns. The majority of the features employed in off-line signature verification research are static features. They produce statistical information using the geometry information of signatures. Very few researchers have attempted recovering dynamic information from static images of signatures. In fact there has been no technique which has fully justified the use of intrinsic rules that govern the formation of the signatures. Although these rules are not yet unveiled, the extraction of features should still concentrate on the facts about the pattern, which really remain unchanged regardless of the perspective. In other words, the features extracted from handwriting should be tolerant to different types of variations and distortion such as rotation, shift, dilation, activation-time, pulse-strength, etc.. Whenever the feature extraction technique is tolerant to variations, the better the pattern is described in the sense that the feature space is made smaller. In smaller feature space, feature vectors extracted from genuine signatures become closer to others. Consequently, it is necessary to identify the available types of invariance lying deep under the physiological processes that produce the signatures or handwriting at large.

Nevertheless, feature extraction techniques must also represent the acquired information in a suitable form so that they can best utilise the capabilities of the learning/classification technique employed. The extracted features may also follow different laws of distribution. Therefore the representations of the extracted information do have a certain impact on the accuracy that an approach could achieve. It is necessary to represent the extracted information appropriately so that better verification accuracy can be obtained.

This section is organized into two major sections. The first section (3.3.1) details various local features investigated in this research. This includes three state-of-the-art feature extraction techniques and another three newly proposed techniques. The second section, (3.3.2) is devoted
to describing the global features investigated.

### 3.3.1 Local Features

The three state-of-the-art local feature extraction techniques chosen for detailed investigation in this research are the MDF, Camastra, and Gradient features. They will be analysed in detail in sections 3.3.1.1, 3.3.1.2, and 3.3.1.3 respectively. Some proposed modifications, including a minor modification to the MDF, are also described. Apart from those, three newly proposed local feature extraction techniques, which are Variance, Gaussian Grid, and Curvature Map will be described one by one in sections 3.3.1.4 through 3.3.1.6. Amongst these feature extraction technique, MDF is a structural feature, whilst others are grid-based features.

#### 3.3.1.1 Modified Direction Feature

The first local feature extraction technique, which is investigated in depth in the present research, is the Modified Direction Feature (MDF). MDF reportedly produced encouraging results in both cursive character recognition as well as signature verification. Therefore, it is of interest to examine this technique in detail in order to improve its performance or develop an enhanced feature extraction technique.

The MDF is created by combining the transition features (TFs) \[63\] and the direction feature (DFs) \[26\]. By simultaneously capturing the location and direction from the same transitions pixels, the MDF is supposed to produce better experimental results compared to either the TFs or the DFs employed individually.

Blumenstein et al \[24\] reported accuracies up to 81.58\% for the upper-case cursive handwritten character recognition problem. In this work, the RBF-NNs with a Gaussian activation function were employed. In signature verification, Armand et al. \[12\] reported relatively high accuracies of up to 89.61\% can be obtained using the MDF with RBF-NNs. These authors employed an early version of the GPDS corpus which consisted of 39 signature sets. The peak performance of 91.21\% was obtained when additional global features were integrated. The MDF has also shown encouraging results in other pattern recognition problems. In \[67\], it was employed to identify people in beach scenes captured by webcams. In \[68\], the MDF was used to detect illicit objects for aviation security.

MDF utilises the spatial and direction information at each point on the contour. It examines the pattern’s contour in all four directions and extracts the location of transitions (LT) and the direction at the transition (DT), specifically transitions from background to foreground. The LT and DT values are subsequently grouped and averaged to obtain a smaller feature vector. In \[32\], Camastra commented that the local averaging process produces values which are tolerant to noise and variations which is useful in character recognition. An accuracy of 90\% was reportedly obtained with SVMs despite the remarkably small dimension (34) of the feature vectors in that research. However, the application of local averaging may be inappropriate as signatures are more complicated.

Despite the same rapid handwriting movement procedures in cursive handwriting and signatures, the movement in signatures is much more sophisticated and skilled \[86\]. Consequently, it is necessary to perform experiments to verify the optimal reported configuration for an enhanced version of MDF. This includes the fine tuning of the following parameters: (1) number of transitions, (2) number of horizontal segments, (3) and number of vertical segments.
Originally, the MDF employed the contour boundaries of patterns as its inputs [23]. It is suggested that, the replacement of the boundaries with partially recovered trajectories would help MDF extract the inner structural information more precisely and consequently improve verification accuracy.

**Direction Feature**  The MDF makes use of the direction of movement of each contour segment. As described in [217], the DFs values are extracted from the contour of the object in the following procedure:

1. Locate starting point and intersection point  
   As in English cursive handwriting, a letter usually starts from the lower left hand side, the first foreground pixel in the lower left hand side is chosen as the starting point. Intersection points are determined as those foreground pixels that have more than two foreground pixel neighbours.

2. Distinguishing individual contour segments  
   Once a contour loop has been located, it is traversed and the foreground pixels are assigned with direction codes. The code assigned to the foreground pixels is updated whenever changes in the moving direction occurs. Altogether, a contour segment is identified based on the following rules:
   (a) A corner condition is satisfied when the direction of traversal changes $90^\circ$ from the left diagonal direction to the right diagonal direction or vice versa.
   (b) During segment detection, when direction changes occur more than three times
   (c) The previous direction has been continuously the same and the length of the previous direction is greater than three pixels

3. Labelling line segment information  
   In this step, the foreground pixels are assigned with direction values as the contour segments are traversed. The codes for possible traversal directions include vertical, right diagonal, horizontal, and left diagonal are 2, 3, 4, and 5 respectively.

4. Line type normalization  
   In this step, the dominant direction value is computed for each contour segment. Here the dominant direction of a contour segment is defined by the direction value, which appears most frequently in that segment. This value is subsequently assigned to every point of the contour segment.

**Location of Transitions**  The other information utilized by the MDF is the locations where transitions between background and foreground pixels occur (LTs). This information is computed simply by dividing the distance traversed along the direction being examined by the corresponding dimension of the input image.

**Feature Values Normalisation**  The direction at the location of a transition is recorded together with the Location Transition. The DT values are scaled to the range between 0 and 1 by
After the LT and DT values have been computed for each point on the contours, the values on each layer are grouped together and averaged in a predefined number of groups in order to obtain a feature vector of fixed dimension. In the original implementation of the MDF, the number of rows or columns in each group is the largest integer that is smaller than one fifth of the size of the dimension being inspected. In images where the width or height are not divisible by 5, there are up to 4 columns or rows, which are unaccounted for. The lower the resolution of the images, the higher the impact on accuracies. It is expected that the missing information will cause negative impact on the performance of the MDF feature.

Algorithm 3.5 New grouping strategy for the MDF

Require: \textit{dimension} Segment\{i.e. width or height\}

Require: \textit{GROUP} Segment
1: \texttt{size} ← \textit{dimension}/\textit{GROUP}
2: \texttt{for} \texttt{group} = 0...\textit{GROUP} − 1 \texttt{do}
3: \hspace{1em} \texttt{from} ← \|\textit{group} \times \texttt{size}||
4: \hspace{1em} \texttt{to} ← \|(\textit{group} + 1) \times \texttt{size}||
5: \hspace{1em} \texttt{for} \texttt{index} = \texttt{from}...\texttt{to}-1 \texttt{do}
6: \hspace{2em} ...
7: \hspace{1em} \texttt{end for}
8: \texttt{end for}

A small modification is proposed to enable the employment of all the rows and columns in the present research. In the first step, the dimension (\texttt{size}) of each group is computed by dividing the dimension being inspected by the number of the group. The indices of the first and last elements in the group \textit{i} are round((\textit{i} − 1) \times \texttt{size}) and round(\textit{i} \times \texttt{size}) − 1 respectively. The pseudo-code of the updated algorithm is presented in Algorithm 3.5. To avoid ambiguity, from this point onward the newly modified MDF variant will be referred as \textit{e-MDF} (\textit{e} for equal).

3.3.1.2 Camastra Feature

The second local feature extraction technique investigated in this research is the one proposed for cursive character recognition by Camastra [32]. This is a relatively light weight grid-based feature extraction technique which consists of a core 32D local feature and two global features: aspect
ratio \((\frac{\text{width}}{\text{height}})\) and the relative position of the baseline to the character itself. Thornton et al. [208] reported that this feature combination outperformed the MDF feature in the cursive handwritten character recognition problem [208]. In the present research, the performance of the core feature, namely the Camstra Feature, will be investigated in the context of signature verification.

In its recommended configuration for cursive character recognition, the input image should be zoned using a \(4 \times 4\) grid. The width and height of each grid cell are the rounded up values of corresponding dimension divided by 4:

\[
 w = \left\lceil \frac{\text{imgwidth}}{4} \right\rceil \tag{3.3.1}
\]

and

\[
 h = \left\lceil \frac{\text{imgheight}}{4} \right\rceil \tag{3.3.2}
\]

where \(\lceil \cdot \rceil\) denotes the ceiling operator.

Whilst the first three rows or columns of the grid are placed in a non-overlapping fashion, the last two columns or rows of cells may be overlapped by up to 3 pixels where the width or height is not divisible by 4. The impact of the overlap can be considered insignificant in images having large dimensions.

Once located, two distinct values are extracted from each cell of the grid. The first one is density, which is the proportion between the number of foreground pixels \(n_i\) in each cell and the area of a grid cell:

\[
 f_1 = \frac{n_i}{\text{cellwidth} \times \text{cellheight}} \tag{3.3.3}
\]

The other feature value is the difference between the sums of the second order of the number of pixels in the horizontal and vertical directions:

\[
 f_2 = \frac{1}{2} \left( 1 + \frac{1}{hw^2} \sum_{i=1}^{w} n_i^2 - \frac{1}{h^2w} \sum_{j=1}^{w} n_j^2 \right) \tag{3.3.4}
\]

where \(h\) and \(w\) are the width and height of each cell, respectively. In total, the local feature proposed by Camstra consisted of \(4 \times 4 \times 2 = 32\) feature values.

In the present research, the optimal configuration for signature verification of the Camstra feature was found experimentally by comparing the AERs obtained using various grid dimensions. The number of grid lines in the horizontal and vertical directions of the grid were kept identical. Although the baseline feature had been integrated in the original work of Camstra, this information is not readily available in signatures. It is noted that the lack of a baseline feature may potentially be a factor that deteriorates the performance of the Camstra Feature for signature verification.

### 3.3.1.3 Gradient Feature

The Gradient feature is another grid-based feature extraction technique investigated in this research in addition to the MDF and Camstra features. This feature was first introduced by Wakabayashi et al. [220] in 1995 for the cursive character recognition problem. Similar to the MDF, the Gradient feature also employs directional and spatial information extracted from the writing. However, the representation of the information in the Gradient feature is quite different from the MDF.
 CHAPTER 3. RESEARCH METHODOLOGY AND PROPOSED TECHNIQUES

It is expected that the comparative performance analyses of the MDF and Gradient features would provide an insight into the construction of more advanced feature extraction techniques. As described in [154], the extraction of the Gradient Feature is performed in the following steps:

**Step 1:** A $2 \times 2$ mean filtering is applied 5 times on the input image.

**Step 2:** The grey scale image obtained in Step 1 is normalized so that the mean grey scale becomes zero with a maximum value of 1.

**Step 3:** The normalized image is then segmented into $17 \times 7$ blocks. Compromising trade-off between accuracy and complexity, this block size is determined experimentally. To get the bounding box of the grey-scale image, the image is converted into two-tone using Otsu’s thresholding algorithm [158]. This will exclude unnecessary background information from the image.

**Step 4:** A Roberts filter is then applied on the image to obtain the gradient image. The arc tangent of the gradient (direction of gradient) is quantized into 32 directions and the strength of the gradient $f(x; y)$ is defined as follows:

$$f(x, y) = \sqrt{(\Delta u)^2 + (\Delta v)^2}$$  \hspace{1cm} (3.3.5)

and the direction of gradient $\theta(x, y)$ is:

$$\theta(x, y) = \tan^{-1} \frac{\Delta v}{\Delta u}$$

where

$$\Delta u = g(x + 1, y + 1) - g(x, y)$$

and

$$\Delta v = g(x + 1, y) - g(x, y + 1)$$

and $g(x, y)$ is the grey level of $(x, y)$ point.

**Step 5:** The histograms of the values of 32 quantized directions are computed for each of the $17 \times 7$ blocks.

**Step 6:** To further reduce the size of the feature vector whilst minimizing the distortion of directional information, the $17 \times 7$ blocks directional histogram is down sampled into $9 \times 4$ blocks and 16 directions using $5 \times 5$ Gaussian filters. The $\sigma$ parameter of the filters was set to be 0.9. Finally, a $9 \times 4 \times 16 = 576$ dimensional feature vector is obtained.

Figure 3.3.2 illustrates a signature and its directional histogram.

---

**3.3.1.4 Variance Feature**

The mean and variance are important measures in probability and statistics. The variance describes how far numbers of a set are distributed from the mean value. It is noted that in signature patterns, the mean and variance extracted from adjacent rows or columns are generally approximate. A
sudden change in both mean and variance often indicates the existence of one or more strokes. It is predicted that an appropriate representation of these values could produce encouraging verification accuracies in signature verification.

Algorithm 3.6 Variance Feature Extraction

Require: Binary image \( I(x, y) \)

Require: Number of groups in the horizontal \( G_{hor} \) and vertical \( G_{ver} \) directions.

1: for \( x = 1 \ldots \text{width} \) do
2: \( V_{x}^{\text{col}} \leftarrow \text{var}(y_i : I(x, y_i) = 1) \)
3: \( M_{x}^{\text{col}} \leftarrow \text{mean}(y_i : I(x, y_i) = 1) \)
4: end for
5: for \( y = 1 \ldots \text{height} \) do
6: \( V_{y}^{\text{row}} \leftarrow \text{var}(x_i : I(x_i, y) = 1) \)
7: \( M_{y}^{\text{row}} \leftarrow \text{mean}(x_i : I(x_i, y) = 1) \)
8: end for
9: \( \text{size}_{\text{hor}} \leftarrow \text{width} / G_{\text{hor}} \)
10: \( \text{size}_{\text{ver}} \leftarrow \text{height} / G_{\text{ver}} \)
11: for \( i = 1 \ldots G_{\text{hor}} \) do
12: \( f_{H}^{\text{mean}} \leftarrow \sum_{i=1}^{i} \text{size}_{\text{hor}} M_{i}^{\text{col}} \)
13: \( f_{H}^{\text{var}} \leftarrow \sum_{i=1}^{i} \text{size}_{\text{hor}} V_{i}^{\text{col}} \)
14: end for
15: for \( j = 1 \ldots G_{\text{ver}} \) do
16: \( f_{V}^{\text{mean}} \leftarrow \sum_{j=1}^{j} \text{size}_{\text{ver}} M_{j}^{\text{col}} \)
17: \( f_{V}^{\text{var}} \leftarrow \sum_{j=1}^{j} \text{size}_{\text{ver}} V_{j}^{\text{col}} \)
18: end for
19: return \( \{f_{H}^{\text{mean}}, f_{H}^{\text{var}}, f_{V}^{\text{mean}}, f_{V}^{\text{var}}\} \)

To compute the Variance Feature, the mean and variance are first computed for each row and column. After that, the adjacent values of each type are grouped together and the average values are calculated for each group. The final feature vector is created by rearranging the average values obtained in the previous step. The purpose of the averaging procedure is to create feature vectors with a predefined dimension. The whole process is presented in Algorithm 3.6.

3.3.1.5 Gaussian Grid Feature

In a recent investigation [154], the performance of the Modified Direction Feature (MDF) and the Gradient feature has been compared using the same settings. The experimental results showed that the Gradient feature outperformed the MDF. Interestingly, both the MDF and the Gradient feature employ local and directional information of signature contours. The major difference is the way information was represented.

The MDF extracts directional information from the normalized contour whilst in the Gradient feature this information is extracted in more detail using a number of quantized directions (32). The location information, where from directional information has been sampled, has also been utilized by both techniques. The Gradient feature encoded this information in the row and column id of each element of the gradient matrix. MDF recorded the location of the transition as LT values. To reduce the feature vector dimension, the MDF employed local averaging whilst the Gradient feature employed the Gaussian filter. One major difference between these two feature extraction techniques, that was observed, is the blurring process employed by the Gradient feature. This operation was performed by applying a \( 2 \times 2 \) mean filter on the input image 5 times.
Blurring not only helps smooth/repair broken contour segments but it also emitted information, i.e. a high frequency signal, from one point to its surrounding area. By doing this, further image manipulation can benefit from more stable input and the stability against small variations can be maintained. It is hypothesized that the blurring process had significant impact on the performance of the Gradient feature and largely contributed to its enhanced performance compared to the MDF. The above observations lead us to the development of the Gaussian Grid feature extraction technique.

The Gaussian Grid feature employs the signature contours as its input. From the contour representation of a signature image, the Gaussian Grid feature extraction technique performs the following steps:

**Step 1:** The input signature contour image is divided into \( m \times n \) equal tiles.

**Step 2:** By tracing the contours in each block the 4-direction chain code histogram [191, 112, 113] of each block is created. Every step from a pixel to its adjacent one of the four directions (horizontal, vertical, left-diagonal, and right-diagonal) are counted. There are four matrices of size \( m \times n \) for each direction, namely \( H \), \( V \), \( L \), and \( R \).

**Step 3:** Apply a Gaussian smoothing filter to each directional \( m \times n \) matrix \( A \) obtained in the previous step.

\[
A_{ij}^* = \sum_{d_i=-\infty}^{\infty} \sum_{d_j=-\infty}^{\infty} A_{i+d_i,j+d_j} \frac{1}{2\pi\sigma^2} e^{-\frac{d_i^2+d_j^2}{2\sigma^2}} 
\]  

(3.3.6)

**Step 4:** The value of each element of each matrix obtained in the previous step is adjusted by dividing its value by the maximum value of the four matrices.

\[
A_{ij} = \frac{A_{ij}}{\max(H_{xy}; V_{xy}; L_{xy}; R_{xy})} 
\]  

(3.3.7)

Figure 3.3.3c illustrates the combined matrix after being filtered and normalized in this step. In this figure as well as Figure 3.3.3b, the colours represent directions whilst their luminance represents the accumulated chain code value after being normalized.

**Step 5:** From the two-matrix pairs horizontal \((H)\) and vertical \((V)\) matrices, left-diagonal \((L)\) and right-diagonal \((R)\) matrices, two new matrices \(\oplus\) and \(\otimes\) are established using the following two equations:
CHAPTER 3. RESEARCH METHODOLOGY AND PROPOSED TECHNIQUES

62

Figure 3.3.4: The normalized curvature map of a signature. Dark spots are regions with very high curvature

\[ \oplus_{ij} = \begin{cases} \min(H_{ij}, V_{ij}) & \text{max}(H_{ij}, V_{ij}) \neq 0 \\ \max(H_{ij}, V_{ij}) = 0 & \text{max}(H_{ij}, V_{ij}) \neq 0 \end{cases} \] (3.3.8)

\[ \otimes_{ij} = \begin{cases} \min(L_{ij}, R_{ij}) & \text{max}(L_{ij}, R_{ij}) \neq 0 \\ \max(L_{ij}, R_{ij}) = 0 & \text{max}(L_{ij}, R_{ij}) \neq 0 \end{cases} \] (3.3.9)

As reported in an earlier publication [152], it was determined that the proportions between the accumulated histogram value of perpendicular directions are relatively stable features and produce encouraging results. It is expected that the inclusion of the \( \oplus \) and \( \otimes \) matrices to the proposed feature would provide the classifiers with additional useful information so that overall accuracies could be increased.

**Step 6**: Finally, the feature vector is formed by rearranging the six matrices \( H, V, L, R, \oplus, \text{ and } \otimes \). The dimension of the output feature vector is \( m \times n \times 6 \).

3.3.1.6 Curvature Map Feature

In Section 3.2, a trajectory recovery framework was proposed. However, any feature extraction technique that employs the recovered trajectory must consider the fact that erroneous trajectory recovery results are inevitable due to the complex nature of the signatures. The Curvature Map is a newly proposed feature extraction technique that utilises the recovered contour segments and provides the classifier with additional information about the spatial distribution of stroke curvature in signature images.

Once the contour segments are recovered, the pseudo-curvature value \( C(x, y) \) is computed for each point \( I(x, y) \) on the contour. Several 12 \( \times \) 12 curvature maps \( (M) \) are employed and record the location of the pseudo-curvature values. Each curvature map \( i \) : \( 1 \leq i \leq M \) is responsible for a range of curvature: \( C_{i, \text{lower}} \leq C(x, y) \leq C_{i, \text{upper}} \). Initially, all the cells of the maps are assigned values of 0. As the pseudo-curvature is computed for a contour element \( I(x, y) \), the value of the cell \( \left( \left\lfloor \frac{x}{\text{cell width}} \right\rfloor, \left\lfloor \frac{y}{\text{cell height}} \right\rfloor \right) \) of the corresponding curvature map is reassigned the value of 1. After every element of the contours have been considered, M maps consisting of 0 and 1 values are obtained. Finally, the Gaussian filter is applied on the maps as previously described in the Gaussian Grid Feature sub-section. The output feature vector is created by rearranging the values of the maps. Figure 3.3.4 depicts the normalized curvature map of a signature computed using its recovered trajectory.
3.3.2 Global Features

Although random forgeries can be detected easily, it has been reported that this type of forgery accounts for more than 90% of the detected cases [136, 53]. Therefore a signature verification system must be capable of identifying random forgeries reliably. Many researchers [10] believed this could be achieved by employing global features in addition to the local features.

Global feature extraction techniques treat input images as a whole and compute features at a global scale. Each dimension of the feature vector of a global feature can only be computed after the whole pattern has been considered. As they produce an overall impression of the image, it is believed that global features should enhance the accuracy of the verification system by rejecting random forgeries effectively.

This sub-section presents an insight into some conventional global investigated in the field prior to describing the newly-proposed global features including Ratio, Energy, Trajectory Length, and Moment-based features.

3.3.2.1 Ratio Feature

Researchers in signature verification, or pattern recognition at large, have proposed and investigated a large number of feature extraction techniques. Many of those attempt to achieve better verification rate by combining two or more distinct features. The search for a better performance subset using available features is an open problem [64] and is not included in the scope of the current research. Apart from that, it is believed that improvement can also be achieved by appropriately representing the input information.

To test the hypothesis that the representation of information does have a certain impact on the accuracy in signature verification, experiments with two distinct representations of a simple feature, (aspect) Ratio, were conducted in this research. The Ratio feature describes the relation between the width and the height of the rectangle that encloses a signature image. As the random forgeries are produced without knowledge about the shape and size as well as width and height of genuine signatures, the Ratio feature should help distinguish random forgeries effectively.

The first ratio formula using the width and height dimensions is the one which was employed in signature verification experiments using neural networks by Armand et al. [11, 12] and Blumenstein et al. [25]. This formula was originally proposed by Liu [132] for cursive character recognition. From the width and height of a signature image, the angle between the right diagonal and the left edge of the image is computed before being normalized by $\pi/2$.

$$R = \frac{\text{arctan}(\text{width/height})}{\pi/2} \quad (3.3.10)$$

Given that one dimension, width or height, of the signature image is fixed then the other...
dimension would likely follow a normal distribution. The usage of the arctangent function would project data unevenly near the corresponding mean value \( (R_{mean}) \), causing the computation of the universal decision threshold less stable. The newly proposed formula addresses this issue by simply dividing the minimum of the width and height values by their maximum:

\[
R^2 = \frac{\min\{width, height\}}{\max\{width, height\}}
\]  
(3.3.11)

Both formulas 3.3.10 and 3.3.11 produce feature values between 0 and 1 but have distinct probability distribution functions. Figure 3.3.5 illustrates the angle which is employed in by Equation 3.3.10.

3.3.2.2 Energy Feature

The second newly proposed global feature in the present research is the Energy Feature. This global feature provides an overall impression of the total energy consumed by movements in each of the following directions: horizontal, vertical, left-diagonal, and right-diagonal. As the rotation in genuine signatures is relatively small, the Energy Feature can be considered stable against rotation.

The Energy Feature was inspired by the hypothesis that the total energy authentic writers use to produce their signatures is relatively stable. The preprogrammed signing process supposedly consumes the same amount of energy each time a signature is created and this information can be approximated from the digital image of a signature.

To compute the Energy feature, every contour segment of the signature image is traversed. As the contour segments are traversed, the number of movements in each of the four main directions is counted. The feature values that form the final feature vector are the width/height divided by horizontal/vertical energy, the proportion between the minimum and the maximum of the horizontal and vertical energy, the proportion between the minimum and the maximum of the left-diagonal and right-diagonal energy, and the proportion between the minimum and the maximum of the sum of horizontal and vertical energy and the sum of the left-diagonal and right-diagonal energy. Algorithm 3.7 details the computation of the Energy Feature.

3.3.2.3 Trajectory Length Feature

The first truly rotation invariant feature investigated in this research is Trajectory Length. This global feature utilises information derived from the length of the pen movement/trajectory.

The most notable characteristic of the length of the trajectory of a signature or handwriting is its invariability with respect to rotation. Moreover, trajectory length is a representation of the total amount of energy consumed during the production of a signature, which is believed to be relatively stable between writing sessions. The length of the trajectory can be approximated with high precision directly from the trajectory of the pen. Unfortunately this information is not readily available in signature images. In the present work, trajectory length is approximated by dividing the total length of all contours by 2.

The trajectory length feature described above is inspired by the Image Area feature employed by Papamarkos and Balzakis in their signature verification research [161]. In that work, the Image Area feature was computed by simply counting the number of the foreground pixels included in the skeleton representation of the signature. The advantage of trajectory length extraction using the contour boundary over using the skeleton is simplicity and accuracy. Skeletonisation often requires
Algorithm 3.7 Energy Feature

function Energy(binary image im)
   1: Sum_h ← 0 {horizontal energy}
   2: Sum_v ← 0 {vertical energy}
   3: Sum_L ← 0 {energy from the left diagonal}
   4: Sum_R ← 0 {energy from the right diagonal}
   5: h ← im.height
   6: w ← im.width
   7: Mark all black pixels of im as unvisited
   8: for each black pixel p of im do
      9: if p is not visited then
         10: track(p)
      11: end if
   12: end for
   13: e_v ← h/Sum_v
   14: e_h ← w/Sum_h
   15: e_hv ← min(Sum_v, Sum_h)/max(Sum_v, Sum_h)
   16: e_LR ← min(Sum_L, Sum_R)/max(Sum_L, Sum_R)
   17: e_v ← Sum_v + Sum_h
   18: e_r ← min(e_hv, e_LR)/max(e_hv, e_LR)
   19: return {e_v, e_h, e_hv, e_LR, e_r}
end function

procedure track(pixel p)
   1: mark p as visited
   2: for each neighbour pixel pNb of p do
      3: if pNb is not visited then
         4: {subscripts y and x mean the row and column of a pixel}
         5: Sum_v ← Sum_v + |pNb_y - p_y|
         6: Sum_h ← Sum_h + |pNb_x - p_x|
         7: if (pNb_y - p_y) * (pNb_x - p_x) = 1 then
            8: Sum_L ← Sum_L + 1
         9: end if
      10: if (pNb_y - p_y) * (pNb_x - p_x) = -1 then
         11: Sum_R ← Sum_R + 1
      12: end if
      13: track(pNb)
      14: break
   15: end if
   16: end for
end procedure

Figure 3.3.6: Line segments of unequal length represented by the same amount of pixels as an effect of digitisation
Algorithm 3.8 Trajectory length

1: \( \text{ADJUST\_ODD} \leftarrow \{\sqrt{(i+1)^2 + 1} - i|i=0..\text{MAX\_INDEX}\} \)
2: \( \text{ADJUST\_EVEN} \leftarrow \{\sqrt{(i+1)^2 + i^2} - i\sqrt{2}|i=0..\text{MAX\_INDEX}\} \)
3: length \( \leftarrow 0 \)
4: for unvisited point \((x,y)\) on the contour do
5: \hspace{1em} direction \( \leftarrow 0 \)
6: \hspace{1em} count \( \leftarrow 0 \)
7: \hspace{1em} restart \( \leftarrow \) true
8: \hspace{1em} while (stepCode \( \leftarrow \) next\((x,y)\) \( \geq 0\)) do
9: \hspace{2em} if restart then
10: \hspace{3em} if stepCode is odd then
11: \hspace{4em} length \( \leftarrow \) length + \( \sqrt{2} \)
12: \hspace{3em} else
13: \hspace{4em} length \( \leftarrow \) length + 1
14: \hspace{3em} end if
15: \hspace{3em} count \( \leftarrow 1 \)
16: \hspace{3em} direction \( \leftarrow \) stepCode
17: \hspace{3em} restart \( \leftarrow \) false
18: \hspace{2em} else
19: \hspace{3em} if stepCode = direction \( \pm 45^\circ \) then
20: \hspace{4em} if count \( \geq \) MAX\_INDEX then
21: \hspace{5em} count \( \leftarrow \) MAX\_INDEX - 1
22: \hspace{4em} end if
23: \hspace{3em} if stepCode is odd then
24: \hspace{4em} length \( \leftarrow \) length + ADJUST\_ODD[count]
25: \hspace{3em} else
26: \hspace{4em} length \( \leftarrow \) length + ADJUST\_EVEN[count]
27: \hspace{3em} end if
28: \hspace{3em} restart \( \leftarrow \) true
29: \hspace{2em} end else
30: \hspace{3em} if stepCode is odd then
31: \hspace{4em} length \( \leftarrow \) length + \( \sqrt{2} \)
32: \hspace{3em} else
33: \hspace{4em} length \( \leftarrow \) length + 1
34: \hspace{3em} end if
35: \hspace{3em} if stepCode = direction then
36: \hspace{4em} increase count \( \leftarrow + 1 \)
37: \hspace{3em} else
38: \hspace{4em} count \( \leftarrow 1 \)
39: \hspace{4em} end if
40: \hspace{3em} end if
41: \hspace{2em} end if
42: \hspace{1em} Mark point \((x,y)\) as visited
43: \hspace{1em} Update new position \((x,y)\)
44: \hspace{1em} end while
45: end for
46: return length/2
more computation than contour extraction and it produces artefacts. Besides, line segments of different length can be represented by the same number of foreground pixels due to the effect of digitisation as illustrated in Figure 3.3.6. In order to obtain a more accurate approximation of trajectory length an adapted version of Harrington’s curve length approximation algorithm [76] was employed. The pseudo-code for the computation of this feature is presented in Algorithm 3.8.

3.3.2.4 Moment Feature

Apart from the Trajectory Length feature, the Moment feature is another rotation invariant global feature investigated in this research. The coordinates of the geometric centre or centre of gravity \( g(x_0, y_0) \) of an image \( I(x, y) \) is the pair of values \( x_0 = \frac{m_{10}}{m_{00}}, y_0 = \frac{m_{01}}{m_{00}} \) where \( m_{pq} \) denotes the \((p, q)\)-order of the moment:

\[
m_{pq} = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} x^p y^q I(x, y) dx dy
\]

The centre of gravity is fixed in relation to any point of the set. The distance between any points and the geometric centre does not vary when the signature image rotates. In other words, these distances are rotation invariant.

Two feature values of the Moment feature are computed as follows:

\[
f_1 = \sum_{-\infty}^{\infty} \sum_{-\infty}^{\infty} \sqrt{(x - x_0)^2 + (y - y_0)^2} I(x, y)
\]

\[
f_2 = \sum_{-\infty}^{\infty} \sum_{-\infty}^{\infty} \frac{1}{\sqrt{(x - x_0)^2 + (y - y_0)^2}} I(x, y)
\]

The first feature value is the sum of the distances from every black pixel to the moment centre and the second feature value is the sum of the inversions of those distances. It can be seen that the magnitude of \( f_1 \) is rather sensitive to outliers whose distance from the centre of gravity is relatively large whilst \( f_2 \) is not. These feature values are then normalized using predefined constants to obtain values within the range \([0..1]\). As the shape and size of genuine signatures of an individual are relatively stable, it is expected that their centre of gravity as well as the above features have stable values and can be used to distinguish genuine signatures from the forgeries.
Chapter 4

EXPERIMENTAL RESULTS

This chapter presents the relevant experimental results that were obtained using the techniques proposed in Chapter 3. The majority of signature verification experiments were conducted using Support Vector Machines for classification. The kernel function employed was the RBF kernel. For experimentation with each set of features, the gamma parameter of the RBF kernel was tuned so that the lowest average error rate could be obtained. In addition, the trade-off between training error and margin was fixed at 1000. The RBF and RBP neural networks were only employed in the cursive character recognition experiments. The results obtained from each experiment or set of relevant experiments are shown in tabular form and are preceded with a brief explanation. The results presented usually include the highest and most significant, or those that assist the recognition of trends in the data. It should be noted that the results reported in the following sections are only a small proportion of the numerous results obtained throughout the course of this research.

The organisation of Chapter 4 is as follows: Firstly, Section 4.1 presents the results of the proposed trajectory recovery technique. The results for local features follows in Section 4.2. After that, the experimental results obtained using global features are presented in Section 4.3. Finally, Section 4.4 present results of the fusion of features. The results presented in this chapter will be further analysed and discussed in Chapter 5.

4.1 Results of the Proposed Trajectory Recovery Framework

4.1.1 Results of the Curvature Maxima Locator

The ultimate objective of the Curvature Maxima Locator Algorithm (CMLA) is to find, from the contours, every point where pairs of distinct strokes meet. Similar to other algorithms involving the computation of curvature from discrete contours, the CMLA suffers from the imprecise curvature values computed. Consequently, potentially missing one or more curvature maxima is inevitable.

To evaluate the performance of the CMLA, it is best to employ a dual-mode database so that the results can be verified using the on-line data. One such database is the MCYT database. However, due to the scope of this work, the relatively small database US-SIGBASE [149], was employed instead. The results of this 51 signature corpus were examined manually. The signatures with the curvature maxima found by the CMLA are presented in tables from 4.1.1 to 4.1.4. In
CHAPTER 4. EXPERIMENTAL RESULTS

Table 4.1: Results of the Curvature Maxima Locator Obtained Using the US-SIGBASE dataset

<table>
<thead>
<tr>
<th>Missing Maxima</th>
<th>Samples</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>19</td>
</tr>
<tr>
<td>1</td>
<td>7</td>
</tr>
<tr>
<td>2</td>
<td>14</td>
</tr>
<tr>
<td>3+</td>
<td>11</td>
</tr>
</tbody>
</table>

those figures, the missing maxima, which are essential for the reconstruction of the true trajectory, are marked by tiny solid circles. Table 4.1 summarises the results obtained using the CMLA.

4.1.2 Bridging Scores Computation

After the signature contours have been segmented, the bridging scores are computed for each potential pair of segments. The outcome of this process was not final and therefore its performance was not assessed in terms of accuracy. Figure 4.1.5 illustrates a signature with fitting circles at stroke intersections. It can be noted that the fitting objects are mostly circles. The fitting objects are lines only when every point of either member of the matching pair has been used.

4.1.3 Trajectory Estimation

Generally speaking, if every curvature maximum resulting from the intersection of strokes on the contour could be located, there exists a bridging configuration that reflects the true trajectory of the pen. However, there are cases where one or more curvature maxima could not be located correctly. In signatures with incorrect or missing curvature maxima points, the proposed trajectory estimation technique still produces a matching configuration. The impact of the such errors can be considered local. The results obtained can still be used for feature extraction. In the signature images in tables from 4.1.1 to 4.1.4, the erroneous choices computed by the proposed framework are marked with solid black lines.

4.2 Results of Local Features

This section reports the experimental results for the local features described in Section 3.3.1. The first section, Section 4.2.1, presents the results for the MDF Feature and its variants. It is further divided into two sub-sections to accommodate results acquired with the 75dpi and 300dpi versions of the GPDS-160 corpus. Subsequently, signature verification results using the Camastra Feature and results facilitating the comparison of the MDF Feature and the Camastra Feature are presented in Section 4.2.2. The performance of the Gradient Feature in conjunction with SVMs for signature verification is reported in Section 4.2.3. The results of the Variance Feature and the Gaussian Grid Feature follows in Section 4.2.4 and Section 4.2.5, respectively. Finally, the results for the Curvature Map feature are presented in Section 4.2.6.

4.2.1 Results Obtained using the Modified Direction Feature and its variants

In the present research, many of the experiments with the MDF feature were conducted using both 75dpi and 300dpi versions of the GPDS-160 corpus. The results of the MDF Feature therefore will
Figure 4.1.1: Missing POIs - 01 to 12
Figure 4.1.2: Missing POIs - 13 to 24
Figure 4.1.3: Missing POIs - 25 to 36
Figure 4.1.4: Missing POIs - 37 to 51
be presented in three separate sections. The first two sections 4.2.1.1 and 4.2.1.2 present the results obtained with the original and enhanced MDF using the 75dpi corpus. The last section (4.2.1.3) presents the results obtained using the 300dpi corpus and the enhanced variants of the MDF.

4.2.1.1 Results Obtained using the MDF for the 75dpi GPDS-160 Corpus

Table 4.2: Experimental Results of the MDF Feature for the 75dpi Database

<table>
<thead>
<tr>
<th>σ</th>
<th>Group</th>
<th>FRR</th>
<th>FAR1</th>
<th>FAR2</th>
<th>AER</th>
</tr>
</thead>
<tbody>
<tr>
<td>6.6e-1</td>
<td>5</td>
<td>17.09%</td>
<td>0.12%</td>
<td>19.61%</td>
<td>18.35%</td>
</tr>
<tr>
<td>7.6e-1</td>
<td>5</td>
<td>18.14%</td>
<td>0.11%</td>
<td>18.32%</td>
<td>18.23%</td>
</tr>
<tr>
<td>7.8e-1</td>
<td>5</td>
<td>18.59%</td>
<td>0.11%</td>
<td>17.92%</td>
<td>18.26%</td>
</tr>
<tr>
<td>8.0e-1</td>
<td>5</td>
<td>19.02%</td>
<td>0.11%</td>
<td>17.52%</td>
<td>18.27%</td>
</tr>
</tbody>
</table>

As previously described in Section 3.3.1.1, MDF is a sophisticated feature extraction technique. Its performance can be adjusted by varying a number of parameters. In their research, Blumenstein et al. [24] suggested that the LTs and DTs values extracted in each of the four directions (left-right, right-left, top-down, bottom-up) should be grouped into 5 groups. In addition, it was recommended that up to 3 transitions should be employed in order to obtain the best results. These settings were recommended for the cursive character recognition problem and may be inappropriate if being applied directly to signature verification. Therefore, an experiment using the MDF Feature with the proposed settings was conducted. Table 4.2 presents the results of this experiment.

Table 4.3: Experimental Results of the MDF with 2 Transitions (MDF_L12)

<table>
<thead>
<tr>
<th>σ</th>
<th>FRR</th>
<th>FAR1</th>
<th>FAR2</th>
<th>AER</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.20</td>
<td>17.13%</td>
<td>0.16%</td>
<td>19.79%</td>
<td>18.46%</td>
</tr>
<tr>
<td>1.25</td>
<td>17.62%</td>
<td>0.16%</td>
<td>19.24%</td>
<td>18.43%</td>
</tr>
<tr>
<td>1.30</td>
<td>18.17%</td>
<td>0.15%</td>
<td>18.75%</td>
<td>18.46%</td>
</tr>
<tr>
<td>1.32</td>
<td>18.40%</td>
<td>0.15%</td>
<td>18.50%</td>
<td>18.45%</td>
</tr>
<tr>
<td>1.35</td>
<td>18.73%</td>
<td>0.14%</td>
<td>18.19%</td>
<td>18.46%</td>
</tr>
</tbody>
</table>

In the present research, many of the preliminary experiments were conducted using the 75dpi database. Table 4.3 presents the results of the experiment using a variant of the MDF in which only the first two transitions were recorded in order to assess the significance of the 3rd transition. These results suggest that the discrimination power of the transitions on the inner layers might be not as strong as those on the outer layers.
CHAPTER 4. EXPERIMENTAL RESULTS

Figure 4.2.1: The performance of the original MDF in conjunction with other global features

Figure 4.2.2: The performance of the original MDF in conjunction with other global features

Other preliminary experiments using the original MDF in conjunction with other global features, including Ratio (R), New Ratio (R2), and Energy (E5), were also conducted. The AERs obtained from these experiments are summarized in Figure 4.2.1.

4.2.1.2 Results Obtained Using the e-MDF for the 75dpi GPDS-160 Corpus

As previously discussed in Section 3.3.1.1, the original implementation of the MDF may discard up to 4 rows or columns of the input image where the dimensions are not divisible by the number of groups. An enhancement was subsequently proposed to employ all of the available data. The results presented in this section were obtained using the enhanced version of the MDF.

Table 4.4: Experimental Results of the e-MDF Feature

<table>
<thead>
<tr>
<th>σ</th>
<th>Group</th>
<th>Size</th>
<th>FRR</th>
<th>FAR1</th>
<th>FAR2</th>
<th>AER</th>
</tr>
</thead>
<tbody>
<tr>
<td>9.00e-1</td>
<td>5</td>
<td>120</td>
<td>19.11%</td>
<td>0.1%</td>
<td>16.39%</td>
<td>17.75%</td>
</tr>
<tr>
<td>0.32</td>
<td>9</td>
<td>216</td>
<td>18.47%</td>
<td>0.09%</td>
<td>16.69%</td>
<td>17.58%</td>
</tr>
<tr>
<td>1.83e-1</td>
<td>12</td>
<td>288</td>
<td>17.90%</td>
<td>0.10%</td>
<td>17.90%</td>
<td>17.90%</td>
</tr>
</tbody>
</table>

At the resolution of 75dpi, the height of some signature images are only 12 pixels. This restricts the number of groups to be no more than 12. The e-MDF would cease to function properly whenever the width or height dimension is smaller than the number of groups. Table 4.4 presents the experimental results obtained using the enhanced MDF with varying numbers of groups.

In order to evaluate the impact of the proposed enhancement more reliably, additional experi-
ments were conducted. These experiments employed feature sets similar to those reported in the previous section (4.2.1.1). Figure 4.2.2 reports the AERs of these experiments.

4.2.1.3 Results Obtained using the Modified Direction Feature on the 300dpi GPDS-160 Corpus

The previous section has presented the experimental results of the MDF using the 75dpi GPDS-160 signature corpus. In this section, some experiments were rerun using the 300dpi resolution version of the same signature corpus. These experiments assist the evaluation of the impact of acquisition resolution on verification accuracies.

Table 4.5: Experimental Results of the MDF Feature using the 300dpi GPDS-160 Corpus

<table>
<thead>
<tr>
<th>σ</th>
<th>Group Size</th>
<th>FRR</th>
<th>FAR1</th>
<th>FAR2</th>
<th>AER</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.16</td>
<td>5</td>
<td>120</td>
<td>18.63%</td>
<td>0.09%</td>
<td>20.30%</td>
</tr>
<tr>
<td>0.43</td>
<td>9</td>
<td>216</td>
<td>17.78%</td>
<td>0.09%</td>
<td>20.53%</td>
</tr>
<tr>
<td>2.9e-1</td>
<td>12</td>
<td>288</td>
<td>18.48%</td>
<td>0.07%</td>
<td>19.41%</td>
</tr>
<tr>
<td>2.2e-1</td>
<td>15</td>
<td>360</td>
<td>19.23%</td>
<td>0.06%</td>
<td>18.42%</td>
</tr>
<tr>
<td>1.68e-1</td>
<td>18</td>
<td>432</td>
<td>19.11%</td>
<td>0.07%</td>
<td>18.58%</td>
</tr>
<tr>
<td>1.4e-1</td>
<td>21</td>
<td>504</td>
<td>19.61%</td>
<td>0.07%</td>
<td>17.98%</td>
</tr>
<tr>
<td>8.8e-2</td>
<td>30</td>
<td>720</td>
<td>19.88%</td>
<td>0.07%</td>
<td>17.58%</td>
</tr>
</tbody>
</table>

The high resolution 300dpi GPDS-160 database facilitates experiments with an increased number of groups compared to its 75dpi counterpart. Table 4.5 presents the results employing the MDF with a different number of group values.

Table 4.6: Experimental Results of the MDF Feature with the 1st and 2nd layers of transition using the 300dpi GPDS-160 Corpus

<table>
<thead>
<tr>
<th>σ</th>
<th>FRR</th>
<th>FAR1</th>
<th>FAR2</th>
<th>AER</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.90</td>
<td>17.95%</td>
<td>0.2%</td>
<td>21.19%</td>
<td>19.57%</td>
</tr>
<tr>
<td>1.94</td>
<td>18.2%</td>
<td>0.20%</td>
<td>20.93%</td>
<td>19.56%</td>
</tr>
<tr>
<td>1.98</td>
<td>18.49%</td>
<td>0.20%</td>
<td>20.69%</td>
<td>19.59%</td>
</tr>
<tr>
<td>2.02</td>
<td>18.79%</td>
<td>0.19%</td>
<td>20.40%</td>
<td>19.59%</td>
</tr>
<tr>
<td>2.06</td>
<td>19.04%</td>
<td>0.19%</td>
<td>20.16%</td>
<td>19.6%</td>
</tr>
<tr>
<td>2.2</td>
<td>20.14%</td>
<td>0.18%</td>
<td>19.29%</td>
<td>19.71%</td>
</tr>
<tr>
<td>2.4</td>
<td>21.65%</td>
<td>0.16%</td>
<td>18.06%</td>
<td>19.85%</td>
</tr>
<tr>
<td>2.5</td>
<td>22.44%</td>
<td>0.15%</td>
<td>17.44%</td>
<td>19.94%</td>
</tr>
</tbody>
</table>

Table 4.7: Experimental Results of the MDF Feature with the 2nd and 3rd layers of transition using the 300dpi GPDS-160 Corpus

<table>
<thead>
<tr>
<th>σ</th>
<th>FRR</th>
<th>FAR1</th>
<th>FAR2</th>
<th>AER</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.40</td>
<td>17.12%</td>
<td>0.12%</td>
<td>23.03%</td>
<td>20.07%</td>
</tr>
<tr>
<td>1.48</td>
<td>17.86%</td>
<td>0.11%</td>
<td>22.19%</td>
<td>20.02%</td>
</tr>
<tr>
<td>1.50</td>
<td>18.06%</td>
<td>0.11%</td>
<td>21.94%</td>
<td>20.00%</td>
</tr>
<tr>
<td>1.54</td>
<td>18.51%</td>
<td>0.10%</td>
<td>21.48%</td>
<td>19.99%</td>
</tr>
<tr>
<td>1.58</td>
<td>18.94%</td>
<td>0.10%</td>
<td>21.04%</td>
<td>19.99%</td>
</tr>
<tr>
<td>1.6</td>
<td>19.17%</td>
<td>0.10%</td>
<td>20.83%</td>
<td>20.00%</td>
</tr>
<tr>
<td>1.8</td>
<td>21.47%</td>
<td>0.08%</td>
<td>18.79%</td>
<td>20.13%</td>
</tr>
<tr>
<td>1.9</td>
<td>22.61%</td>
<td>0.07%</td>
<td>17.75%</td>
<td>20.18%</td>
</tr>
</tbody>
</table>

Tables 4.6, 4.7, and 4.8 present the results from the MDF variants in which one of the layers
of transitions is removed. Compared to the original configuration with 120 feature values, the feature vectors produced by these variants are smaller with 80 feature values. The experiments with these variants of the MDF were conducted using the 300dpi version of the GPDS-160 corpus. These results provide information about the contribution of each layer of transition to the overall performance of the recommended settings of the MDF.

An experiment with an MDF variant (MDF_L1), which includes only the 1st transitions, was also conducted. The results from these experiments as well as those from the experiments with the original MDF and MDF_L12 help evaluate the impact of the inclusion of the higher order transitions into the MDF. The results of this experiment is presented in Table 4.9.

Table 4.10 summarises the average error rates of the experiments with MDF variants having 5 groups and different transition settings.

The experimental results presented in tables from 4.2 to 4.10 were conducted by varying the number of segments or number of transitions separately. Although better performance is unlikely to be achieved by employing more than 3 transitions, an experiment was conducted to check if any progress could be made by increasing both the number of groups and the number of transitions. In

Table 4.10: The best AERs of MDF with different layer settings obtained using the 300dpi database

<table>
<thead>
<tr>
<th>Layer of Transition Employed</th>
<th>Dimension</th>
<th>AER</th>
</tr>
</thead>
<tbody>
<tr>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
</tbody>
</table>
Table 4.11: Experimental Results of a variant of the MDF Feature with 15 groups and 5 transitions

<table>
<thead>
<tr>
<th>$\sigma$</th>
<th>FRR</th>
<th>FAR1</th>
<th>FAR2</th>
<th>AER</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.0e-1</td>
<td>17.99%</td>
<td>0.05%</td>
<td>20.73%</td>
<td>19.36%</td>
</tr>
<tr>
<td>1.1e-1</td>
<td>19.67%</td>
<td>0.05%</td>
<td>19.06%</td>
<td>19.37%</td>
</tr>
<tr>
<td>1.125e-1</td>
<td>20.07%</td>
<td>0.05%</td>
<td>18.57%</td>
<td>19.32%</td>
</tr>
<tr>
<td>1.175e-1</td>
<td>20.94%</td>
<td>0.04%</td>
<td>17.75%</td>
<td>19.35%</td>
</tr>
<tr>
<td>1.225e-1</td>
<td>21.85%</td>
<td>0.04%</td>
<td>16.91%</td>
<td>19.38%</td>
</tr>
<tr>
<td>1.275e-1</td>
<td>22.76%</td>
<td>0.04%</td>
<td>16.07%</td>
<td>19.42%</td>
</tr>
</tbody>
</table>

Figure 4.2.3: The performance of the e-MDF feature with varying number of groups using the 75dpi and 300dpi databases

In this experiment, the number of groups was set at 15, and up to 5 transitions were employed. The dimensions of this MDF variant was $15 \text{ groups} \times 5 \text{ transitions} \times 4 \text{ directions} \times 2 \text{ features} = 600$. Table 4.11 presents the results obtained from this experiment. Due to the unsatisfying results attained, no other experiments with the MDF were conducted using more than 3 transitions.

Figure 4.2.3 concludes the section for the results of the MDF feature extraction technique by summarising the results of various group sizes using the two databases.

### 4.2.2 Results Obtained Using the Camasta Feature

Table 4.12: Recognition accuracies of various feature sets for the C-Cube Corpus

<table>
<thead>
<tr>
<th>Centres / Hidden Unit</th>
<th>{MDF, R}</th>
<th>{MDF, {R, E5} }</th>
<th>{MDF_L12, Camasta} (Split-A)</th>
<th>Camasta (Split-B)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1024</td>
<td>71.93%</td>
<td>72.39%</td>
<td>82.30%</td>
<td>79.30%</td>
</tr>
<tr>
<td>2048</td>
<td>75.78%</td>
<td>76.17%</td>
<td>84.94%</td>
<td>82.19%</td>
</tr>
<tr>
<td>4096</td>
<td>78.95%</td>
<td>79.39%</td>
<td>86.80%</td>
<td>83.84%</td>
</tr>
<tr>
<td>5120</td>
<td>79.92%</td>
<td>80.52%</td>
<td>87.15%</td>
<td>84.27%</td>
</tr>
<tr>
<td>64</td>
<td>72.49%</td>
<td>72.79%</td>
<td>77.50%</td>
<td>73.46%</td>
</tr>
<tr>
<td>128</td>
<td>76.63%</td>
<td>76.42%</td>
<td>81.14%</td>
<td>77.61%</td>
</tr>
<tr>
<td>256</td>
<td>77.61%</td>
<td>76.95%</td>
<td>83.53%</td>
<td>80.16%</td>
</tr>
<tr>
<td>320</td>
<td>77.82%</td>
<td>78.19%</td>
<td>84.22%</td>
<td>80.74%</td>
</tr>
<tr>
<td>512</td>
<td>*4.41%</td>
<td>8.94%</td>
<td>5.30%</td>
<td>1.56%</td>
</tr>
</tbody>
</table>
As previously mentioned, the Camastra Feature was created for cursive character recognition. The performance of the Camastra Feature in that area was evaluated using the C-Cube dataset [32, 130, 208] in conjunction with the SVM, RBF, and RBP neural networks. The results for RBF and RBP is displayed in the last two columns of Table 4.12. Details about the Split-A and Split-B variants of the C-Cube dataset can be found in [208].

Table 4.13: Experimental Results using variants of the Camastra Feature

<table>
<thead>
<tr>
<th>Grid Configuration</th>
<th>Size</th>
<th>σ</th>
<th>FRR</th>
<th>FAR1</th>
<th>FAR2</th>
<th>AER</th>
</tr>
</thead>
<tbody>
<tr>
<td>4 × 4</td>
<td>32</td>
<td>40</td>
<td>19.89%</td>
<td>0.23%</td>
<td>20.77%</td>
<td>20.33%</td>
</tr>
<tr>
<td>8 × 8</td>
<td>128</td>
<td>2.1e1</td>
<td>20.4%</td>
<td>0.07%</td>
<td>18.29%</td>
<td>19.35%</td>
</tr>
<tr>
<td>12 × 12</td>
<td>288</td>
<td>4.4</td>
<td>24.63%</td>
<td>0.13%</td>
<td>17.77%</td>
<td>21.2%</td>
</tr>
<tr>
<td>16 × 16</td>
<td>512</td>
<td>4</td>
<td>24.07%</td>
<td>0.15%</td>
<td>18.36%</td>
<td>21.21%</td>
</tr>
</tbody>
</table>

It is generally agreed that the width and height of signature images are much greater than cursive characters. Signatures are also more sophisticated with a larger number of strokes, intersections, and overlaps. Therefore if the dimensions of the grid are too small, the horizontal-vertical second order difference feature value (Equation 3.3.4) of each cell extracted by the Camastra Feature will vary greatly, and will negatively affect verification accuracies. Therefore experiments with variants of the Camastra feature having larger grid dimensions 8 × 8, 12 × 12, and 16 × 16 were conducted in the current research. Table 4.13 presents the results of these experiments.

Figure 4.2.4 concludes this section by depicting the FRR and FAR2 curves of the variants of the Camastra Feature as a function of the gamma parameter of the RBF kernel.
4.2.3 Results Obtained Using the Gradient Feature

The optimal settings of the Gradient Feature were determined experimentally using the Squared Mahalanobis Distance (SMD) classifiers [154] with the assistance of researchers from Mie University: Yumiko Kawazoe and Tetsushi Wakabayashi. The training and testing processes in those experiments were performed without random forgeries. The results obtained using the optimal configuration in conjunction with SMD classifiers are presented in Table 4.14. Table 4.15 presents the results obtained using SVMs. The 300dpi database was employed in these experiments.

### Table 4.14: Experimental Results for the Gradient Feature Obtained Using Squared Mahalanobis Distance classifiers

<table>
<thead>
<tr>
<th>Threshold</th>
<th>FRR</th>
<th>FAR1</th>
<th>FAR2</th>
<th>AER</th>
</tr>
</thead>
<tbody>
<tr>
<td>-3182</td>
<td>11.39</td>
<td>23.46</td>
<td>16.77%</td>
<td></td>
</tr>
<tr>
<td>-3176</td>
<td>12.48</td>
<td>21.74</td>
<td>16.61%</td>
<td></td>
</tr>
<tr>
<td>-3171</td>
<td>13.44</td>
<td>20.41</td>
<td>16.55%</td>
<td></td>
</tr>
<tr>
<td>-3164</td>
<td>14.80</td>
<td>18.63</td>
<td>16.52%</td>
<td></td>
</tr>
<tr>
<td>-3160</td>
<td>15.76</td>
<td>17.66</td>
<td>16.62%</td>
<td></td>
</tr>
<tr>
<td>-3156</td>
<td>16.71</td>
<td>16.84</td>
<td>16.78%</td>
<td></td>
</tr>
<tr>
<td>-3153</td>
<td>17.45</td>
<td>16.23</td>
<td>16.92%</td>
<td></td>
</tr>
</tbody>
</table>

### Table 4.15: Experimental Results for the Gradient Feature Obtained Using SVMs

<table>
<thead>
<tr>
<th>σ</th>
<th>FRR</th>
<th>FAR1</th>
<th>FAR2</th>
<th>AER</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.41</td>
<td>13.84%</td>
<td>0.02%</td>
<td>16.7%</td>
<td>15.27%</td>
</tr>
<tr>
<td>4.3e-1</td>
<td>14.81%</td>
<td>0.02%</td>
<td>15.53%</td>
<td>15.17%</td>
</tr>
<tr>
<td>4.4e-1</td>
<td>15.26%</td>
<td>0.02%</td>
<td>14.95%</td>
<td>15.10%</td>
</tr>
<tr>
<td>4.65e-1</td>
<td>16.54%</td>
<td>0.02%</td>
<td>13.51%</td>
<td>15.03%</td>
</tr>
<tr>
<td>4.8e-1</td>
<td>17.41%</td>
<td>0.02%</td>
<td>12.76%</td>
<td>15.09%</td>
</tr>
<tr>
<td>4.9e-1</td>
<td>18.02%</td>
<td>0.02%</td>
<td>12.25%</td>
<td>15.13%</td>
</tr>
<tr>
<td>5.0e-1</td>
<td>18.62%</td>
<td>0.02%</td>
<td>11.73%</td>
<td>15.17%</td>
</tr>
</tbody>
</table>

The dimension of the Variance Feature vector is controlled by only one parameter which is the number of segments of the input image in the horizontal or the vertical directions. In the experiments with this feature extraction technique, the number of segments was varied between 6 and 18 with a step of 3. Table 4.16 displays the results of the Variance Feature with varying number of segments.

### Table 4.16: Experimental Results for the Variance Feature

<table>
<thead>
<tr>
<th>Segment</th>
<th>Feature Values</th>
<th>σ</th>
<th>FRR</th>
<th>FAR1</th>
<th>FAR2</th>
<th>AER</th>
</tr>
</thead>
<tbody>
<tr>
<td>6</td>
<td>24</td>
<td>8.0</td>
<td>23.03%</td>
<td>0.75%</td>
<td>21.37%</td>
<td>22.20%</td>
</tr>
<tr>
<td>9</td>
<td>36</td>
<td>4.2</td>
<td>22.82%</td>
<td>0.6%</td>
<td>20.95%</td>
<td>21.89%</td>
</tr>
<tr>
<td>12</td>
<td>48</td>
<td>3.4</td>
<td>25.22%</td>
<td>0.45%</td>
<td>18.49%</td>
<td>21.85%</td>
</tr>
<tr>
<td>15</td>
<td>60</td>
<td>2.0</td>
<td>23.82%</td>
<td>0.52%</td>
<td>19.91%</td>
<td>21.87%</td>
</tr>
<tr>
<td>18</td>
<td>72</td>
<td>1.8</td>
<td>25.69%</td>
<td>0.47%</td>
<td>18.5%</td>
<td>22.10%</td>
</tr>
</tbody>
</table>

The results obtained with the Variance Feature are relatively encouraging due to the small dimensions of the feature vector. The 6 segment variant with only 24 feature dimensions produced
CHAPTER 4. EXPERIMENTAL RESULTS

21.6
21.8
22
22.2
22.4
4  6  8  10  12  14  16  18  20
AER (%)
Number of segments

Figure 4.2.5: The AERs of the Variance Feature with Different Number of Segments

Table 4.17: Experimental Results Obtained Using the $9 \times 9$ Gaussian Grid Feature

<table>
<thead>
<tr>
<th>$\sigma$</th>
<th>FRR</th>
<th>FAR1</th>
<th>FAR2</th>
<th>AER</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.6e-1</td>
<td>8.47%</td>
<td>0.09%</td>
<td>23.42%</td>
<td>15.94%</td>
</tr>
<tr>
<td>0.7e-1</td>
<td>9.62%</td>
<td>0.07%</td>
<td>20.84%</td>
<td>15.23%</td>
</tr>
<tr>
<td>0.8e-1</td>
<td>11.03%</td>
<td>0.06%</td>
<td>18.6%</td>
<td>14.81</td>
</tr>
<tr>
<td>0.9e-1</td>
<td>12.6%</td>
<td>0.05%</td>
<td>16.36%</td>
<td>14.48%</td>
</tr>
<tr>
<td>1.0e-1</td>
<td>14.37%</td>
<td>0.04%</td>
<td>14.42%</td>
<td>14.39%</td>
</tr>
<tr>
<td>1.1e-1</td>
<td>16.53%</td>
<td>0.03%</td>
<td>12.54%</td>
<td>14.53%</td>
</tr>
<tr>
<td>1.2e-1</td>
<td>18.96%</td>
<td>0.03%</td>
<td>10.75%</td>
<td>14.85%</td>
</tr>
<tr>
<td>1.3e-1</td>
<td>21.46%</td>
<td>0.02%</td>
<td>9.2%</td>
<td>15.33%</td>
</tr>
<tr>
<td>1.4e-1</td>
<td>24.27%</td>
<td>0.02%</td>
<td>7.89%</td>
<td>16.08%</td>
</tr>
</tbody>
</table>

an AER of 22.20%. Figure 4.2.5 summarises the AERs of these experiments.

4.2.5 Results Obtained Using the Gaussian Grid Feature

There are three parameters needed to be fine tuned for the Gaussian Grid Feature to reach its peak performance. They are the horizontal and vertical dimensions of the grid and the sigma parameter of the 2D Gaussian filter. In the experiments using the Gaussian Grid Feature, the horizontal and vertical dimensions of the grid were equally set.

Similar to other grid-based feature extraction techniques, the dimensions of the grid have to be experimentally determined so that the best results could be achieved. It was decided to employ

Table 4.18: Experimental Results Obtained Using the $12 \times 12$ Gaussian Grid Feature

<table>
<thead>
<tr>
<th>$\sigma$</th>
<th>FRR</th>
<th>FAR1</th>
<th>FAR2</th>
<th>AER</th>
</tr>
</thead>
<tbody>
<tr>
<td>3.3e-2</td>
<td>10.05%</td>
<td>0.04%</td>
<td>19.34%</td>
<td>14.69%</td>
</tr>
<tr>
<td>4.1e-2</td>
<td>12.59%</td>
<td>0.03%</td>
<td>15.39%</td>
<td>13.99%</td>
</tr>
<tr>
<td>4.3e-2</td>
<td>13.38%</td>
<td>0.03%</td>
<td>14.49%</td>
<td>13.93%</td>
</tr>
<tr>
<td>4.5e-2</td>
<td>14.18%</td>
<td>0.02%</td>
<td>13.68%</td>
<td>13.93%</td>
</tr>
<tr>
<td>5.0e-2</td>
<td>16.53%</td>
<td>0.02%</td>
<td>11.68%</td>
<td>14.10%</td>
</tr>
<tr>
<td>5.2e-2</td>
<td>17.54%</td>
<td>0.02%</td>
<td>10.99%</td>
<td>14.26%</td>
</tr>
<tr>
<td>5.4e-2</td>
<td>18.59%</td>
<td>0.01%</td>
<td>10.27%</td>
<td>14.43%</td>
</tr>
<tr>
<td>5.6e-2</td>
<td>19.72%</td>
<td>0.01%</td>
<td>9.59%</td>
<td>14.65%</td>
</tr>
<tr>
<td>5.8e-2</td>
<td>20.88%</td>
<td>0.01%</td>
<td>8.99%</td>
<td>14.93%</td>
</tr>
</tbody>
</table>
Table 4.19: Experimental Results Obtained Using the $15 \times 15$ Gaussian Grid Feature

<table>
<thead>
<tr>
<th>$\sigma$</th>
<th>FRR</th>
<th>FAR1</th>
<th>FAR2</th>
<th>AER</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.80e-2</td>
<td>10.06%</td>
<td>0.04%</td>
<td>19.75%</td>
<td>14.905%</td>
</tr>
<tr>
<td>2.00e-2</td>
<td>11.01%</td>
<td>0.03%</td>
<td>17.97%</td>
<td>14.49%</td>
</tr>
<tr>
<td>2.20e-2</td>
<td>12.07%</td>
<td>0.03%</td>
<td>16.28%</td>
<td>14.175%</td>
</tr>
<tr>
<td>2.45e-2</td>
<td>13.58%</td>
<td>0.02%</td>
<td>14.27%</td>
<td>13.935%</td>
</tr>
<tr>
<td>2.50e-2</td>
<td>13.96%</td>
<td>0.02%</td>
<td>13.91%</td>
<td>13.945%</td>
</tr>
<tr>
<td>2.55e-2</td>
<td>14.32%</td>
<td>0.02%</td>
<td>13.53%</td>
<td><strong>13.925%</strong></td>
</tr>
<tr>
<td>2.60e-2</td>
<td>14.69%</td>
<td>0.02%</td>
<td>13.16%</td>
<td>13.925%</td>
</tr>
<tr>
<td>2.80e-2</td>
<td>16.23%</td>
<td>0.02%</td>
<td>11.89%</td>
<td>14.060%</td>
</tr>
</tbody>
</table>

Table 4.20: Experimental Results Obtained using the Gaussian Grid Feature with varying grid size

<table>
<thead>
<tr>
<th>$\sigma$</th>
<th>Grid size</th>
<th>Dimension</th>
<th>FRR</th>
<th>FAR1</th>
<th>FAR2</th>
<th>AER</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.0e-1</td>
<td>9 $\times$ 9</td>
<td>486</td>
<td>14.37%</td>
<td>0.04%</td>
<td>14.42%</td>
<td>14.39%</td>
</tr>
<tr>
<td>4.5e-2</td>
<td>12 $\times$ 12</td>
<td>864</td>
<td>14.18%</td>
<td>0.02%</td>
<td>13.68%</td>
<td>13.93%</td>
</tr>
<tr>
<td>2.55e-2</td>
<td>15 $\times$ 15</td>
<td>1350</td>
<td>14.32%</td>
<td>0.02%</td>
<td>13.53%</td>
<td>13.925%</td>
</tr>
</tbody>
</table>

the following grid configurations: $9 \times 9$, $12 \times 12$, $15 \times 15$. The results of these experiments are presented in tables 4.17, 4.18, and 4.19, respectively. In these experiments, the sigma parameter of the Gaussian filter was set at 1.2. Table 4.20 summarises the settings and results of these experiments.

Two additional experiments were performed in order to assess the impact of the 2D Gaussian filter on the performance of the Gaussian Grid Feature. In these experiments, the Gaussian filters were not applied on the accumulated direction arrays. The same grid settings, which are $9 \times 9$ and $12 \times 12$, and sigma value were also employed. The results from these two experiments are presented in Table 4.21 and Table 4.22, respectively.

To evaluate the significance of the supplementary local features, an experiment was performed. In this experiment, Step 5 of the feature extraction procedure was bypassed. The $12 \times 12$ grid configuration employed in this experiment produced a $12 \times 12 \times 4 = 576$ dimensional feature vector. Table 4.23 shows the results of this experiment. As expected, the performance of the Gaussian Grid Feature significantly deteriorated as the supplementary matrices had been removed.

4.2.6 Results Obtained Using the Curvature Map Feature

In the experiments using the Curvature Map Feature, the sigma parameter of the Gaussian filter, which was employed to smooth the map, was fixed at 1.2.

Table 4.21: Experimental Results Obtained Using the $9 \times 9$ Directional Map

<table>
<thead>
<tr>
<th>$\sigma$</th>
<th>FRR</th>
<th>FAR1</th>
<th>FAR2</th>
<th>AER</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.0e-2</td>
<td>15.32%</td>
<td>0.08%</td>
<td>23.84%</td>
<td>19.58%</td>
</tr>
<tr>
<td>1.5e-2</td>
<td>16.00%</td>
<td>0.06%</td>
<td>21.91%</td>
<td>18.95%</td>
</tr>
<tr>
<td>2.0e-2</td>
<td>16.88%</td>
<td>0.05%</td>
<td>19.99%</td>
<td>18.43%</td>
</tr>
<tr>
<td>2.5e-2</td>
<td>17.97%</td>
<td>0.04%</td>
<td>18.08%</td>
<td>18.02%</td>
</tr>
<tr>
<td>3.0e-2</td>
<td>19.28%</td>
<td>0.03%</td>
<td>16.31%</td>
<td>17.79%</td>
</tr>
<tr>
<td>3.5e-2</td>
<td>20.82%</td>
<td>0.02%</td>
<td>14.55%</td>
<td><strong>17.68%</strong></td>
</tr>
<tr>
<td>4.0e-2</td>
<td>22.67%</td>
<td>0.02%</td>
<td>12.93%</td>
<td>17.80%</td>
</tr>
<tr>
<td>5.4e-2</td>
<td>24.66%</td>
<td>0.02%</td>
<td>11.37%</td>
<td>18.01%</td>
</tr>
</tbody>
</table>
Table 4.22: Experimental Results Obtained Using the 12 × 12 Directional Map

<table>
<thead>
<tr>
<th>$\sigma$</th>
<th>FRR</th>
<th>FAR1</th>
<th>FAR2</th>
<th>AER</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.0e-3</td>
<td>18.52%</td>
<td>0.09%</td>
<td>22.18%</td>
<td>20.35%</td>
</tr>
<tr>
<td>5.0e-3</td>
<td>19.48%</td>
<td>0.06%</td>
<td>19.84%</td>
<td>19.66%</td>
</tr>
<tr>
<td>7.0e-3</td>
<td>20.12%</td>
<td>0.05%</td>
<td>18.72%</td>
<td>19.42%</td>
</tr>
<tr>
<td>1.1e-2</td>
<td>21.54%</td>
<td>0.04%</td>
<td>16.64%</td>
<td>19.09%</td>
</tr>
<tr>
<td>1.3e-2</td>
<td>22.42%</td>
<td>0.03%</td>
<td>15.56%</td>
<td>18.99%</td>
</tr>
<tr>
<td>1.5e-2</td>
<td>23.39%</td>
<td>0.03%</td>
<td>14.44%</td>
<td>18.91%</td>
</tr>
<tr>
<td>1.7e-2</td>
<td>24.48%</td>
<td>0.02%</td>
<td>13.43%</td>
<td>18.95%</td>
</tr>
<tr>
<td>2.0e-2</td>
<td>26.36%</td>
<td>0.02%</td>
<td>11.96%</td>
<td>19.16%</td>
</tr>
</tbody>
</table>

Table 4.23: Experimental Results Obtained using a Variant of the 12 × 12 Gaussian Grid Feature without Additional Local Features

<table>
<thead>
<tr>
<th>$\sigma$</th>
<th>FRR</th>
<th>FAR1</th>
<th>FAR2</th>
<th>AER</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.3e-1</td>
<td>13.05%</td>
<td>0.05%</td>
<td>18.17%</td>
<td>15.61%</td>
</tr>
<tr>
<td>1.4e-1</td>
<td>14.35%</td>
<td>0.04%</td>
<td>16.68%</td>
<td>15.51%</td>
</tr>
<tr>
<td>1.5e-1</td>
<td>15.89%</td>
<td>0.04%</td>
<td>15.22%</td>
<td>15.55%</td>
</tr>
<tr>
<td>1.6e-1</td>
<td>17.44%</td>
<td>0.04%</td>
<td>13.85%</td>
<td>15.64%</td>
</tr>
</tbody>
</table>

Table 4.24: Experimental Results Obtained Using the 12 × 12 Curvature Map Feature

<table>
<thead>
<tr>
<th>$\sigma$</th>
<th>FRR</th>
<th>FAR1</th>
<th>FAR2</th>
<th>AER</th>
</tr>
</thead>
<tbody>
<tr>
<td>2.0e-1</td>
<td>18.92%</td>
<td>0.11%</td>
<td>21.53%</td>
<td>20.22%</td>
</tr>
<tr>
<td>2.1e-1</td>
<td>19.71%</td>
<td>0.11%</td>
<td>20.69%</td>
<td>20.20%</td>
</tr>
<tr>
<td>2.2e-1</td>
<td>20.62%</td>
<td>0.10%</td>
<td>19.89%</td>
<td>20.25%</td>
</tr>
<tr>
<td>2.3e-1</td>
<td>21.55%</td>
<td>0.09%</td>
<td>19.07%</td>
<td>20.31%</td>
</tr>
<tr>
<td>2.4e-1</td>
<td>22.47%</td>
<td>0.09%</td>
<td>18.29%</td>
<td>20.38%</td>
</tr>
<tr>
<td>2.5e-1</td>
<td>23.44%</td>
<td>0.08%</td>
<td>17.49%</td>
<td>20.46%</td>
</tr>
</tbody>
</table>

Table 4.25: Experimental Results Obtained Using the combination of the Gaussian Grid Feature and the Curvature Map Feature

<table>
<thead>
<tr>
<th>$\sigma$</th>
<th>FRR</th>
<th>FAR1</th>
<th>FAR2</th>
<th>AER</th>
</tr>
</thead>
<tbody>
<tr>
<td>3.92e-2</td>
<td>13.61%</td>
<td>0.02%</td>
<td>13.46%</td>
<td>13.54%</td>
</tr>
<tr>
<td>3.94e-2</td>
<td>13.73%</td>
<td>0.01%</td>
<td>13.35%</td>
<td>13.54%</td>
</tr>
<tr>
<td>3.96e-2</td>
<td>13.84%</td>
<td>0.01%</td>
<td>13.26%</td>
<td>13.55%</td>
</tr>
<tr>
<td>3.98e-2</td>
<td>13.99%</td>
<td>0.01%</td>
<td>13.14%</td>
<td>13.57%</td>
</tr>
<tr>
<td>4.00e-2</td>
<td>14.12%</td>
<td>0.01%</td>
<td>13.05%</td>
<td>13.59%</td>
</tr>
<tr>
<td>4.02e-2</td>
<td>14.26%</td>
<td>0.01%</td>
<td>12.96%</td>
<td>13.61%</td>
</tr>
<tr>
<td>4.06e-2</td>
<td>14.38%</td>
<td>0.01%</td>
<td>12.84%</td>
<td>13.61%</td>
</tr>
</tbody>
</table>
Table 4.26: Experimental Results Obtained Using the $12 \times 12$ Curvature Map Feature Extracted from the Original Contours

<table>
<thead>
<tr>
<th>$\sigma$</th>
<th>FRR</th>
<th>FAR1</th>
<th>FAR2</th>
<th>AER</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.5e-1</td>
<td>17.16%</td>
<td>0.17%</td>
<td>24.11%</td>
<td>20.635%</td>
</tr>
<tr>
<td>1.6e-1</td>
<td>17.87%</td>
<td>0.16%</td>
<td>23.24%</td>
<td>20.555%</td>
</tr>
<tr>
<td>1.7e-1</td>
<td>18.62%</td>
<td>0.14%</td>
<td>22.38%</td>
<td>20.5%</td>
</tr>
<tr>
<td>1.8e-1</td>
<td>19.42%</td>
<td>0.13%</td>
<td>21.53%</td>
<td>20.475%</td>
</tr>
<tr>
<td>1.9e-1</td>
<td>20.24%</td>
<td>0.12%</td>
<td>20.74%</td>
<td>20.49%</td>
</tr>
<tr>
<td>2.0e-1</td>
<td>21.14%</td>
<td>0.11%</td>
<td>19.89%</td>
<td>20.515%</td>
</tr>
<tr>
<td>2.1e-1</td>
<td>22.03%</td>
<td>0.1%</td>
<td>19.04%</td>
<td>20.535%</td>
</tr>
<tr>
<td>2.2e-1</td>
<td>23.02%</td>
<td>0.09%</td>
<td>18.17%</td>
<td>20.595%</td>
</tr>
</tbody>
</table>

Table 4.24 presents the results of the Curvature Map Feature. The performance of the combination including Curvature Map and Gaussian Grid Feature is presented in Table 4.25.

Another experiment was performed in order to evaluate the usefulness of the recovered trajectory. In this experiment, the original contours were employed instead of the recovered trajectory. Table 4.26 presents the results of this experiment.

This concludes the section for the results obtained using local features. In the next section, the results using global features are presented.

### 4.3 Results employing Global Features

Global features consider patterns as a whole. In other words, any components of the feature vector produced by a global feature extraction technique can only be computed after every component of the pattern has been examined. Compared to local feature vectors, the dimensions of local features are often much smaller. Therefore, global features cannot describe patterns in as much detail as local features, and the performance of global features are not expected to be as good as local features. In signature verification, global features are often employed as a supplement to other local features.

The following sections present the experimental results obtained using four global features described in Section 3.3.2 of the previous chapter. Firstly, results of the two ratio formulae are presented in Section 4.3.1. Secondly, Section 4.3.2 displays the results obtained using the Energy feature. This section is followed by Section 4.3.3 with results from the Trajectory Length Feature. Finally, the results of the Moment Feature are given in Section 4.3.4.

#### 4.3.1 Results Obtained Using the Ratio Features

This section presents the experimental results obtained using the two formulae describing the relation between the width and height of signature images. These results indicated that the AERs of the newly proposed formula are generally better than the legacy one in every experiment conducted.

As the computation of both ratio formulae are not affected by the resolution, it is likely that the comparative performance of the feature sets using the ratio formulae do not vary with image resolution. A number of experiments using the ratio formulae were conducted using both the low and high resolution database to check this hypothesis.
CHAPTER 4. EXPERIMENTAL RESULTS

4.3.1.1 Experimental Results of the Ratio Feature Obtained using the 75dpi GPDS-160 Corpus

Table 4.27: Experimental Results of the \{MDF, R2\} feature set for 75dpi GPDS-160 corpus

<table>
<thead>
<tr>
<th>σ</th>
<th>FRR</th>
<th>FAR(^1)</th>
<th>FAR(^2)</th>
<th>AER</th>
</tr>
</thead>
<tbody>
<tr>
<td>7.8e-1</td>
<td>18.15%</td>
<td>0.10%</td>
<td>17.92%</td>
<td>18.04%</td>
</tr>
<tr>
<td>7.9e-1</td>
<td>18.36%</td>
<td>0.10%</td>
<td>17.71%</td>
<td>18.04%</td>
</tr>
<tr>
<td>8.0e-1</td>
<td>18.56%</td>
<td>0.10%</td>
<td>17.49%</td>
<td>18.03%</td>
</tr>
<tr>
<td>8.1e-1</td>
<td>18.76%</td>
<td>0.10%</td>
<td>17.27%</td>
<td>18.02%</td>
</tr>
<tr>
<td>8.2e-1</td>
<td>18.98%</td>
<td>0.10%</td>
<td>17.07%</td>
<td>18.03%</td>
</tr>
<tr>
<td>8.3e-1</td>
<td>19.20%</td>
<td>0.09%</td>
<td>16.88%</td>
<td>18.04%</td>
</tr>
</tbody>
</table>

Table 4.28: Experimental Results of the \{MDF, R\} feature set for 75dpi GPDS-160 corpus

<table>
<thead>
<tr>
<th>σ</th>
<th>FRR</th>
<th>FAR(^1)</th>
<th>FAR(^2)</th>
<th>AER</th>
</tr>
</thead>
<tbody>
<tr>
<td>7.1e-1</td>
<td>17.83%</td>
<td>0.13%</td>
<td>18.92%</td>
<td>18.38%</td>
</tr>
<tr>
<td>7.2e-1</td>
<td>18.02%</td>
<td>0.13%</td>
<td>18.70%</td>
<td>18.36%</td>
</tr>
<tr>
<td>7.3e-1</td>
<td>18.23%</td>
<td>0.13%</td>
<td>18.52%</td>
<td>18.38%</td>
</tr>
<tr>
<td>7.4e-1</td>
<td>18.42%</td>
<td>0.13%</td>
<td>18.29%</td>
<td>18.36%</td>
</tr>
<tr>
<td>7.5e-1</td>
<td>18.64%</td>
<td>0.12%</td>
<td>18.11%</td>
<td>18.38%</td>
</tr>
<tr>
<td>7.6e-1</td>
<td>18.83%</td>
<td>0.12%</td>
<td>17.90%</td>
<td>18.37%</td>
</tr>
</tbody>
</table>

As the Ratio features have only one feature value, it was decided to compare these features indirectly using other features. The total dimension of the chosen feature set should not be too large so that the impact of the Ratio features can be clearly observed. Tables 4.27 and 4.28 present the results for the \{MDF, R\} and \{MDF, R2\} feature sets.

Figure 4.3.1 illustrates the FAR\(^1\) rates of the \{MDF, R\} and \{MDF, R2\} feature sets as a function of the FRR.
CHAPTER 4. EXPERIMENTAL RESULTS

Figure 4.3.2: The FAR1 rates of the \{MDF, R\} and \{MDF, R2\} feature sets as a function of the FRR.

Table 4.29: Experimental Results of the \{MDF, R, E5\} feature set

<table>
<thead>
<tr>
<th>(\sigma)</th>
<th>FRR</th>
<th>FAR1</th>
<th>FAR2</th>
<th>AER</th>
</tr>
</thead>
<tbody>
<tr>
<td>7.0e-1</td>
<td>15.91%</td>
<td>0.10%</td>
<td>19.28%</td>
<td>17.60%</td>
</tr>
<tr>
<td>7.5e-1</td>
<td>16.84%</td>
<td>0.09%</td>
<td>18.21%</td>
<td>17.53%</td>
</tr>
<tr>
<td>7.6e-1</td>
<td>17.05%</td>
<td>0.09%</td>
<td>18.00%</td>
<td>17.53%</td>
</tr>
<tr>
<td>8.1e-1</td>
<td>18.09%</td>
<td>0.09%</td>
<td>16.90%</td>
<td>17.50%</td>
</tr>
<tr>
<td>8.4e-1</td>
<td>18.64%</td>
<td>0.08%</td>
<td>16.47%</td>
<td>17.56%</td>
</tr>
<tr>
<td>8.7e-1</td>
<td>19.29%</td>
<td>0.08%</td>
<td>15.80%</td>
<td>17.55%</td>
</tr>
</tbody>
</table>

Table 4.30: Experimental Results of the \{MDF, R2, E5\} feature set

<table>
<thead>
<tr>
<th>(\sigma)</th>
<th>FRR</th>
<th>FAR1</th>
<th>FAR2</th>
<th>AER</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.72</td>
<td>16.07%</td>
<td>0.09%</td>
<td>18.71%</td>
<td>17.39%</td>
</tr>
<tr>
<td>0.76</td>
<td>16.91%</td>
<td>0.08%</td>
<td>17.83%</td>
<td>17.37%</td>
</tr>
<tr>
<td>0.78</td>
<td>17.38%</td>
<td>0.08%</td>
<td>17.42%</td>
<td>17.40%</td>
</tr>
<tr>
<td>0.80</td>
<td>17.77%</td>
<td>0.08%</td>
<td>16.96%</td>
<td>17.37%</td>
</tr>
<tr>
<td>0.82</td>
<td>18.23%</td>
<td>0.08%</td>
<td>16.56%</td>
<td>17.40%</td>
</tr>
<tr>
<td>0.84</td>
<td>18.68%</td>
<td>0.07%</td>
<td>16.12%</td>
<td>17.40%</td>
</tr>
</tbody>
</table>

Tables 4.29 and 4.30 present the results of the two ratio features when combined with the \{MDF, E5\} feature set.

Figure 4.3.2 illustrates the FAR1 rates of the \{MDF, R\} and \{MDF, R2\} feature sets as a function of the FRR.

Table 4.31: Experimental Results of the \{MDF_L12, R\} feature set

<table>
<thead>
<tr>
<th>(\sigma)</th>
<th>FRR</th>
<th>FAR1</th>
<th>FAR2</th>
<th>AER</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.2</td>
<td>16.32%</td>
<td>0.15%</td>
<td>19.7%</td>
<td>18.01%</td>
</tr>
<tr>
<td>1.4</td>
<td>18.02%</td>
<td>0.13%</td>
<td>17.6%</td>
<td>17.81%</td>
</tr>
<tr>
<td>1.52</td>
<td>19.22%</td>
<td>0.12%</td>
<td>16.35%</td>
<td>17.79%</td>
</tr>
<tr>
<td>1.6</td>
<td>20.02%</td>
<td>0.11%</td>
<td>15.5%</td>
<td>17.76%</td>
</tr>
<tr>
<td>1.72</td>
<td>21.23%</td>
<td>0.1%</td>
<td>14.34%</td>
<td>17.79%</td>
</tr>
<tr>
<td>1.8</td>
<td>22.14%</td>
<td>0.1%</td>
<td>13.59%</td>
<td>17.87%</td>
</tr>
</tbody>
</table>
Table 4.33: Experimental Results of the \{MDF, R\} feature set with the 300dpi database

<table>
<thead>
<tr>
<th>(\sigma)</th>
<th>FRR</th>
<th>FAR1</th>
<th>FAR2</th>
<th>AER</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.0</td>
<td>16.15%</td>
<td>0.12%</td>
<td>22.87%</td>
<td>19.51%</td>
</tr>
<tr>
<td>1.1</td>
<td>17.6%</td>
<td>0.1%</td>
<td>21.29%</td>
<td>19.46%</td>
</tr>
<tr>
<td>1.14</td>
<td>18.15%</td>
<td>0.1%</td>
<td>20.69%</td>
<td>19.42%</td>
</tr>
<tr>
<td>1.22</td>
<td>19.32%</td>
<td>0.09%</td>
<td>19.58%</td>
<td>19.45%</td>
</tr>
<tr>
<td>1.28</td>
<td>20.25%</td>
<td>0.08%</td>
<td>18.75%</td>
<td>19.5%</td>
</tr>
<tr>
<td>1.4</td>
<td>22.25%</td>
<td>0.07%</td>
<td>17.17%</td>
<td>19.71%</td>
</tr>
</tbody>
</table>

Table 4.34: Experimental Results of the \{MDF, R2\} feature set with the 300dpi database

<table>
<thead>
<tr>
<th>(\sigma)</th>
<th>FRR</th>
<th>FAR1</th>
<th>FAR2</th>
<th>AER</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.90</td>
<td>14.67%</td>
<td>0.12%</td>
<td>24.48%</td>
<td>19.58%</td>
</tr>
<tr>
<td>0.95</td>
<td>15.33%</td>
<td>0.11%</td>
<td>23.6%</td>
<td>19.47%</td>
</tr>
<tr>
<td>1.00</td>
<td>15.99%</td>
<td>0.1%</td>
<td>22.76%</td>
<td>19.37%</td>
</tr>
<tr>
<td>1.10</td>
<td>17.33%</td>
<td>0.09%</td>
<td>21.17%</td>
<td>19.25%</td>
</tr>
<tr>
<td>1.15</td>
<td>18.11%</td>
<td>0.08%</td>
<td>20.41%</td>
<td>19.26%</td>
</tr>
<tr>
<td>1.22</td>
<td>19.16%</td>
<td>0.09%</td>
<td>19.43%</td>
<td>19.3%</td>
</tr>
</tbody>
</table>

Table 4.32: Experimental Results of the \{MDF_L12, R2\} feature set

<table>
<thead>
<tr>
<th>(\sigma)</th>
<th>FRR</th>
<th>FAR1</th>
<th>FAR2</th>
<th>AER</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.24</td>
<td>16.04%</td>
<td>0.15%</td>
<td>19.37%</td>
<td>17.71%</td>
</tr>
<tr>
<td>1.36</td>
<td>17.19%</td>
<td>0.13%</td>
<td>18.05%</td>
<td>17.62%</td>
</tr>
<tr>
<td>1.48</td>
<td>18.41%</td>
<td>0.12%</td>
<td>16.71%</td>
<td>17.56%</td>
</tr>
<tr>
<td>1.58</td>
<td>19.38%</td>
<td>0.11%</td>
<td>15.6%</td>
<td>17.49%</td>
</tr>
<tr>
<td>1.68</td>
<td>20.42%</td>
<td>0.1%</td>
<td>14.58%</td>
<td>17.5%</td>
</tr>
<tr>
<td>1.74</td>
<td>21.1%</td>
<td>0.1%</td>
<td>13.97%</td>
<td>17.54%</td>
</tr>
</tbody>
</table>

The encouraging results of MDF_L12 suggest that this MDF variant may produce similar results to the original implementation. Other global features therefore combined with MDF_L12 to verify this hypothesis. Tables 4.31 and 4.32 presents the results of the \{MDF_L12,R\} and \{MDF_L12, R2\} feature sets.

Figure 4.3.3: Performance of Ratio and Ratio 2 with different feature combinations

Figure 4.3.3 concludes this section by presenting the average error rate of various MDF-based feature sets using the two ratio formulae.
CHAPTER 4. EXPERIMENTAL RESULTS

Table 4.35: Experimental Results of the \{Variance6+6, R\} feature set with the 300dpi database

<table>
<thead>
<tr>
<th>$\sigma$</th>
<th>FRR</th>
<th>FAR1</th>
<th>FAR2</th>
<th>AER</th>
</tr>
</thead>
<tbody>
<tr>
<td>6.0e+0</td>
<td>18.41%</td>
<td>0.67%</td>
<td>24.11%</td>
<td>21.26%</td>
</tr>
<tr>
<td>7.0e+0</td>
<td>19.21%</td>
<td>0.61%</td>
<td>23.1%</td>
<td>21.56%</td>
</tr>
<tr>
<td>8.8e+0</td>
<td>20.79%</td>
<td>0.53%</td>
<td>21.19%</td>
<td><strong>20.99%</strong></td>
</tr>
<tr>
<td>9.8e+0</td>
<td>21.8%</td>
<td>0.49%</td>
<td>20.32%</td>
<td>21.06%</td>
</tr>
<tr>
<td>1.1e+1</td>
<td>22.91%</td>
<td>0.45%</td>
<td>19.27%</td>
<td>21.09%</td>
</tr>
<tr>
<td>1.2e+1</td>
<td>23.93%</td>
<td>0.42%</td>
<td>18.41%</td>
<td>21.17%</td>
</tr>
<tr>
<td>1.4e+1</td>
<td>26.14%</td>
<td>0.36%</td>
<td>16.77%</td>
<td>21.46%</td>
</tr>
</tbody>
</table>

Table 4.36: Experimental Results of the \{Variance6+6, R2\} feature set with the 300dpi GPDS-160 corpus

<table>
<thead>
<tr>
<th>$\sigma$</th>
<th>FRR</th>
<th>FAR1</th>
<th>FAR2</th>
<th>AER</th>
</tr>
</thead>
<tbody>
<tr>
<td>4.0</td>
<td>17.39%</td>
<td>0.8%</td>
<td>25.05%</td>
<td>20.22%</td>
</tr>
<tr>
<td>6.4</td>
<td>19.42%</td>
<td>0.64%</td>
<td>22.38%</td>
<td>20.9%</td>
</tr>
<tr>
<td>7.6</td>
<td>20.49%</td>
<td>0.58%</td>
<td>21.04%</td>
<td>20.77%</td>
</tr>
<tr>
<td>8.4</td>
<td>21.31%</td>
<td>0.54%</td>
<td>20.2%</td>
<td><strong>20.76%</strong></td>
</tr>
<tr>
<td>9.6</td>
<td>22.57%</td>
<td>0.48%</td>
<td>19.04%</td>
<td>20.81%</td>
</tr>
<tr>
<td>1.0</td>
<td>23.02%</td>
<td>0.46%</td>
<td>18.67%</td>
<td>20.85%</td>
</tr>
<tr>
<td>1.2e+1</td>
<td>25.29%</td>
<td>0.38%</td>
<td>16.8%</td>
<td>21.05%</td>
</tr>
</tbody>
</table>

4.3.1.2 Experimental Results of the Ratio Feature Obtained using the 300dpi GPDS-160 Corpus

In the experiments with various modifications of the MDF in the previous section, the results were compared in favour of the R2 feature. Since the computation of the ratio feature is not affected by the resolution of the signature images, those experiments were not conducted again using the 300dpi database.

Table 4.35 and Table 4.36 present the results of the \{Variance6+6, R\} and \{Variance6+6, R2\} feature sets respectively. The $\text{FAR}^1$ rates of these feature sets are a function of the FRR as depicted in Figure 4.3.4.

Table 4.37: Experimental Results of the \{Gaussian Grid, R2\} feature set using the 300dpi GPDS-160 corpus

<table>
<thead>
<tr>
<th>$\sigma$</th>
<th>FRR</th>
<th>FAR1</th>
<th>FAR2</th>
<th>AER</th>
</tr>
</thead>
<tbody>
<tr>
<td>4.0e-2</td>
<td>12.21%</td>
<td>0.04%</td>
<td>15.83%</td>
<td>14.02%</td>
</tr>
<tr>
<td>4.1e-2</td>
<td>12.6%</td>
<td>0.04%</td>
<td>15.36%</td>
<td>13.98%</td>
</tr>
<tr>
<td>4.2e-2</td>
<td>12.98%</td>
<td>0.03%</td>
<td>14.93%</td>
<td>13.96%</td>
</tr>
<tr>
<td>4.3e-2</td>
<td>13.34%</td>
<td>0.03%</td>
<td>14.49%</td>
<td>13.92%</td>
</tr>
<tr>
<td>4.4e-2</td>
<td>13.75%</td>
<td>0.03%</td>
<td>14.02%</td>
<td><strong>13.89%</strong></td>
</tr>
<tr>
<td>4.5e-2</td>
<td>14.19%</td>
<td>0.03%</td>
<td>13.59%</td>
<td>13.89%</td>
</tr>
<tr>
<td>4.6e-2</td>
<td>14.61%</td>
<td>0.03%</td>
<td>13.22%</td>
<td>13.92%</td>
</tr>
</tbody>
</table>

Table 4.37 presents the results obtained using the \{Gaussian Grid, R2\} feature set.

4.3.2 Results Obtained Using the Energy Feature

The Energy Feature extracts information from the signature contours. Its performance can be influenced by the resolution of the input image. In order to evaluate the impact of image resolution
CHAPTER 4. EXPERIMENTAL RESULTS

Figure 4.3.4: The FAR1 rates of the \{Variance6+6, R\} and \{Variance6+6, R2\} feature set as a function of the FRR.

on this feature, many experiments using the Energy feature were conducted using both the 75dpi and 300dpi databases in this research.

4.3.2.1 Results Obtained using the Energy Feature with the 75dpi GPDS-160 Database

Table 4.38: Experimental Results of the Energy Feature using the 75dpi Database

<table>
<thead>
<tr>
<th>σ</th>
<th>FRR</th>
<th>FAR1</th>
<th>FAR2</th>
<th>AER</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>42.57%</td>
<td>1.62%</td>
<td>17.58%</td>
<td>30.08%</td>
</tr>
<tr>
<td>4</td>
<td>37.2%</td>
<td>2.02%</td>
<td>19.76%</td>
<td>28.48%</td>
</tr>
<tr>
<td>4.8</td>
<td>36.28%</td>
<td>2.11%</td>
<td>20.25%</td>
<td>28.27%</td>
</tr>
<tr>
<td>6.8</td>
<td>35.12%</td>
<td>2.22%</td>
<td>20.85%</td>
<td>27.99%</td>
</tr>
<tr>
<td>1.8e1</td>
<td>36.23%</td>
<td>2.39%</td>
<td>20.71%</td>
<td>28.47%</td>
</tr>
<tr>
<td>4.0e1</td>
<td>38.99%</td>
<td>2.17%</td>
<td>18.86%</td>
<td>28.93%</td>
</tr>
<tr>
<td>6.0e1</td>
<td>41.19%</td>
<td>1.98%</td>
<td>17.4%</td>
<td>29.3%</td>
</tr>
</tbody>
</table>

The Energy feature (E5) consists of only 5 feature values. It employs information about the total length of movement in each of the four main directions. Despite having more feature values than the Ratio feature, it is not expected that the performance of the Energy feature will be comparable to the local features. Table 4.38 presents the results of the Energy feature for the 75dpi database.

Table 4.39: Experimental Results of the \{MDF, E5\} Feature Set using the 75dpi Database

<table>
<thead>
<tr>
<th>σ</th>
<th>FRR</th>
<th>FAR1</th>
<th>FAR2</th>
<th>AER</th>
</tr>
</thead>
<tbody>
<tr>
<td>6.8e-1</td>
<td>15.66%</td>
<td>0.10%</td>
<td>19.64%</td>
<td>17.65%</td>
</tr>
<tr>
<td>7.2e-1</td>
<td>16.43%</td>
<td>0.10%</td>
<td>18.68%</td>
<td>17.56%</td>
</tr>
<tr>
<td>7.6e-1</td>
<td>17.20%</td>
<td>0.09%</td>
<td>17.87%</td>
<td>17.54%</td>
</tr>
<tr>
<td>8.0e-1</td>
<td>18.01%</td>
<td>0.09%</td>
<td>16.98%</td>
<td>17.50%</td>
</tr>
<tr>
<td>8.4e-1</td>
<td>18.88%</td>
<td>0.08%</td>
<td>16.20%</td>
<td>17.54%</td>
</tr>
<tr>
<td>8.8e-1</td>
<td>19.78%</td>
<td>0.08%</td>
<td>15.38%</td>
<td>17.58%</td>
</tr>
</tbody>
</table>
CHAPTER 4. EXPERIMENTAL RESULTS

90

Table 4.42: Experimental Results of the {e-MDF, E5} feature set

<table>
<thead>
<tr>
<th>( \sigma )</th>
<th>FRR</th>
<th>FAR1</th>
<th>FAR2</th>
<th>AER</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.0</td>
<td>15.09%</td>
<td>0.09%</td>
<td>22.25%</td>
<td>18.67%</td>
</tr>
<tr>
<td>1.06</td>
<td>15.87%</td>
<td>0.08%</td>
<td>21.23%</td>
<td>18.55%</td>
</tr>
<tr>
<td>1.12</td>
<td>16.8%</td>
<td>0.08%</td>
<td>20.2%</td>
<td>18.5%</td>
</tr>
<tr>
<td>1.18</td>
<td>17.73%</td>
<td>0.07%</td>
<td>19.2%</td>
<td>18.465%</td>
</tr>
<tr>
<td>1.24</td>
<td>18.71%</td>
<td>0.07%</td>
<td>18.31%</td>
<td>18.51%</td>
</tr>
<tr>
<td>1.30</td>
<td>19.73%</td>
<td>0.06%</td>
<td>17.48%</td>
<td>18.605%</td>
</tr>
<tr>
<td>1.32</td>
<td>20.07%</td>
<td>0.06%</td>
<td>17.22%</td>
<td>18.645%</td>
</tr>
</tbody>
</table>

Table 4.40: Experimental Results of the \{MDF\_L12, E5, R2\} feature set

<table>
<thead>
<tr>
<th>( \sigma )</th>
<th>FRR</th>
<th>FAR1</th>
<th>FAR2</th>
<th>AER</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.32</td>
<td>17.35%</td>
<td>0.10%</td>
<td>17.40%</td>
<td>17.38%</td>
</tr>
<tr>
<td>1.34</td>
<td>17.58%</td>
<td>0.10%</td>
<td>17.13%</td>
<td>17.36%</td>
</tr>
<tr>
<td>1.36</td>
<td>17.80%</td>
<td>0.10%</td>
<td>16.88%</td>
<td>17.34%</td>
</tr>
<tr>
<td>1.38</td>
<td>18.32%</td>
<td>0.09%</td>
<td>16.37%</td>
<td>17.35%</td>
</tr>
</tbody>
</table>

In the experiments using the 75dpi database, E5 was employed as a supplement to the MDF Feature. Table 4.39 presents the results using the \{MDF, Energy\} feature set. Only one variant of the MDF, which was MDF\_L12, was chosen for experimentation with the Energy Feature, and the results obtained are presented in Table 4.40. The results of the Energy Feature in conjunction with the \{MDF, R\} and \{MDF, R2\} feature sets were previously reported in Table 4.29 and Table 4.30 respectively, and will not be presented here.

4.3.2.2 Results Obtained using the Energy Feature with the 300dpi GPDS-160 Database

Table 4.41: Experimental Results of the E5 feature

<table>
<thead>
<tr>
<th>( \sigma )</th>
<th>FRR</th>
<th>FAR1</th>
<th>FAR2</th>
<th>AER</th>
</tr>
</thead>
<tbody>
<tr>
<td>4</td>
<td>33.6%</td>
<td>1.86%</td>
<td>22.24%</td>
<td>27.92%</td>
</tr>
<tr>
<td>6</td>
<td>31.11%</td>
<td>2.01%</td>
<td>23.45%</td>
<td>27.28%</td>
</tr>
<tr>
<td>8</td>
<td>30.13%</td>
<td>2.11%</td>
<td>24.11%</td>
<td>27.12%</td>
</tr>
<tr>
<td>1.0e1</td>
<td>29.59%</td>
<td>2.16%</td>
<td>24.34%</td>
<td>26.96%</td>
</tr>
<tr>
<td>1.8e1</td>
<td>30.05%</td>
<td>2.22%</td>
<td>24.51%</td>
<td>27.28%</td>
</tr>
<tr>
<td>4.0e1</td>
<td>32.32%</td>
<td>2.1%</td>
<td>23.13%</td>
<td>27.72%</td>
</tr>
<tr>
<td>6.0e1</td>
<td>33.94%</td>
<td>1.97%</td>
<td>22.13%</td>
<td>28.03%</td>
</tr>
</tbody>
</table>

Table 4.41 presents the experimental results for the Energy Feature (E5) using the 300dpi database. In conjunction with the results from Table 4.38, these results permit the evaluation of the impact of signature image resolution on the Energy Feature.

Experiments employing the Energy Feature in conjunction with other local features were conducted. The results of these experiments are presented in tables from 4.42 to 4.48.

4.3.2.3 Results for Off-line Cursive Character Recognition Using the Energy Feature

In the present research, the performance of the Energy Feature for off-line cursive character recognition was also investigated. The Energy Feature was employed in conjunction with the MDF
Table 4.43: Experimental Results of the \{Camastra8×8, E5\} feature set

<table>
<thead>
<tr>
<th>σ</th>
<th>FRR</th>
<th>FAR1</th>
<th>FAR2</th>
<th>AER</th>
</tr>
</thead>
<tbody>
<tr>
<td>11.6</td>
<td>13.99%</td>
<td>0.11%</td>
<td>19.87%</td>
<td>16.93%</td>
</tr>
<tr>
<td>12.8</td>
<td>14.95%</td>
<td>0.1%</td>
<td>18.82%</td>
<td>16.885%</td>
</tr>
<tr>
<td>13.8</td>
<td>15.88%</td>
<td>0.09%</td>
<td>17.98%</td>
<td>16.93%</td>
</tr>
<tr>
<td>15.0</td>
<td>16.95%</td>
<td>0.08%</td>
<td>16.98%</td>
<td>16.965%</td>
</tr>
<tr>
<td>16</td>
<td>17.98%</td>
<td>0.08%</td>
<td>16.16%</td>
<td>17.07%</td>
</tr>
<tr>
<td>18</td>
<td>20.02%</td>
<td>0.07%</td>
<td>14.72%</td>
<td>17.37%</td>
</tr>
<tr>
<td>20</td>
<td>22.32%</td>
<td>0.06%</td>
<td>13.3%</td>
<td>17.81%</td>
</tr>
</tbody>
</table>

Table 4.44: Experimental Results of the \{Variance6+6, E5\} feature set

<table>
<thead>
<tr>
<th>σ</th>
<th>FRR</th>
<th>FAR1</th>
<th>FAR2</th>
<th>AER</th>
</tr>
</thead>
<tbody>
<tr>
<td>7.0e0</td>
<td>15.58%</td>
<td>0.36%</td>
<td>21.24%</td>
<td>18.41%</td>
</tr>
<tr>
<td>8.0e0</td>
<td>16.99%</td>
<td>0.31%</td>
<td>19.75%</td>
<td>18.37%</td>
</tr>
<tr>
<td>8.2e0</td>
<td>17.29%</td>
<td>0.31%</td>
<td>19.44%</td>
<td>18.365%</td>
</tr>
<tr>
<td>8.8e0</td>
<td>18.11%</td>
<td>0.28%</td>
<td>18.55%</td>
<td>18.33%</td>
</tr>
<tr>
<td>9.6e0</td>
<td>19.22%</td>
<td>0.25%</td>
<td>17.5%</td>
<td>18.36%</td>
</tr>
<tr>
<td>1.1e1</td>
<td>21.45%</td>
<td>0.21%</td>
<td>15.77%</td>
<td>18.61%</td>
</tr>
<tr>
<td>1.2e1</td>
<td>23.26%</td>
<td>0.19%</td>
<td>14.57%</td>
<td>18.915%</td>
</tr>
</tbody>
</table>

Table 4.45: Experimental Results of the \{Variance6+6, R2, E5\} feature set

<table>
<thead>
<tr>
<th>σ</th>
<th>FRR</th>
<th>FAR1</th>
<th>FAR2</th>
<th>AER</th>
</tr>
</thead>
<tbody>
<tr>
<td>6.5</td>
<td>14.66%</td>
<td>0.34%</td>
<td>21.59%</td>
<td>18.13%</td>
</tr>
<tr>
<td>7.0</td>
<td>15.35%</td>
<td>0.32%</td>
<td>20.67%</td>
<td>18.01</td>
</tr>
<tr>
<td>7.5</td>
<td>16.17%</td>
<td>0.3%</td>
<td>19.82%</td>
<td>18.00%</td>
</tr>
<tr>
<td>8.0</td>
<td>16.97%</td>
<td>0.28%</td>
<td>18.97%</td>
<td>17.97%</td>
</tr>
<tr>
<td>8.55</td>
<td>17.98%</td>
<td>0.25%</td>
<td>18.13%</td>
<td>18.06%</td>
</tr>
<tr>
<td>9.0</td>
<td>18.74%</td>
<td>0.24%</td>
<td>17.39%</td>
<td>18.07%</td>
</tr>
<tr>
<td>9.5</td>
<td>19.6%</td>
<td>0.22%</td>
<td>16.62%</td>
<td>18.11%</td>
</tr>
</tbody>
</table>

Table 4.46: Experimental Results of the \{Gradient, E5\} feature set

<table>
<thead>
<tr>
<th>σ</th>
<th>FRR</th>
<th>FAR1</th>
<th>FAR2</th>
<th>AER</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.43</td>
<td>14.65%</td>
<td>0.02%</td>
<td>14.76%</td>
<td>14.705%</td>
</tr>
<tr>
<td>0.45</td>
<td>15.67%</td>
<td>0.02%</td>
<td>13.62%</td>
<td>14.465%</td>
</tr>
<tr>
<td>0.47</td>
<td>16.73%</td>
<td>0.02%</td>
<td>12.46%</td>
<td>14.595%</td>
</tr>
<tr>
<td>0.49</td>
<td>17.95%</td>
<td>0.01%</td>
<td>11.43%</td>
<td>14.69%</td>
</tr>
<tr>
<td>0.50</td>
<td>18.57%</td>
<td>0.01%</td>
<td>10.94%</td>
<td>14.755%</td>
</tr>
<tr>
<td>0.52</td>
<td>19.8%</td>
<td>0.01%</td>
<td>10.01%</td>
<td>14.905%</td>
</tr>
</tbody>
</table>

Table 4.47: Experimental Results of the \{GaussianGrid, E5\} feature set

<table>
<thead>
<tr>
<th>σ</th>
<th>FRR</th>
<th>FAR1</th>
<th>FAR2</th>
<th>AER</th>
</tr>
</thead>
<tbody>
<tr>
<td>3.0e-2</td>
<td>9.31%</td>
<td>0.05%</td>
<td>20.92%</td>
<td>15.115%</td>
</tr>
<tr>
<td>4.2e-2</td>
<td>12.98%</td>
<td>0.03%</td>
<td>14.87%</td>
<td>13.925%</td>
</tr>
<tr>
<td>4.4e-2</td>
<td>13.77%</td>
<td>0.03%</td>
<td>14%</td>
<td>13.885%</td>
</tr>
<tr>
<td>4.5e-2</td>
<td>14.19%</td>
<td>0.03%</td>
<td>13.56%</td>
<td>13.875%</td>
</tr>
<tr>
<td>4.7e-2</td>
<td>15.1%</td>
<td>0.03%</td>
<td>12.73%</td>
<td>13.915%</td>
</tr>
<tr>
<td>4.9e-2</td>
<td>16.02%</td>
<td>0.02%</td>
<td>11.98%</td>
<td>14%</td>
</tr>
</tbody>
</table>
Table 4.48: Experimental Results of the \{CurvatureMap, E5\} feature set

<table>
<thead>
<tr>
<th>$\sigma$</th>
<th>FRR</th>
<th>FAR1</th>
<th>FAR2</th>
<th>AER</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.16</td>
<td>15.09%</td>
<td>0.12%</td>
<td>25.21%</td>
<td>20.15%</td>
</tr>
<tr>
<td>0.18</td>
<td>16.51%</td>
<td>0.1%</td>
<td>23.49%</td>
<td>20%</td>
</tr>
<tr>
<td>0.192</td>
<td>17.39%</td>
<td>0.1%</td>
<td>22.38%</td>
<td>19.885%</td>
</tr>
<tr>
<td>0.204</td>
<td>18.35%</td>
<td>0.09%</td>
<td>21.29%</td>
<td>19.82%</td>
</tr>
<tr>
<td>0.216</td>
<td>19.43%</td>
<td>0.08%</td>
<td>20.31%</td>
<td>19.87%</td>
</tr>
<tr>
<td>0.24</td>
<td>21.65%</td>
<td>0.07%</td>
<td>18.2%</td>
<td>19.925%</td>
</tr>
</tbody>
</table>

Feature and other global features. The database employed in this research was the C-Cube corpus [32] which is publicly available at the URL http://ccc.idiap.ch. This relatively large Latin character corpus is comprised of 38490 and 19133 character images for training and testing respectively.

In these experiments, the RBF and RBP neural networks were employed. Due to memory restrictions, the maximum number of centres used with the RBF networks was limited to 5120, whilst the number of hidden units for the RBP networks were limited to 320.

Figures 4.3.5 and 4.3.6 illustrate the performance of the MDF-R with and without the Energy Feature.
Table 4.49: Experimental Results of the \{MDF, TrajectoryLength\} Feature Set Using the 300dpi Database

<table>
<thead>
<tr>
<th>$\sigma$</th>
<th>FRR</th>
<th>FAR1</th>
<th>FAR2</th>
<th>AER</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.05</td>
<td>16.56%</td>
<td>0.09%</td>
<td>21.99%</td>
<td>19.28%</td>
</tr>
<tr>
<td>1.15</td>
<td>17.96%</td>
<td>0.08%</td>
<td>20.51%</td>
<td>19.24%</td>
</tr>
<tr>
<td>1.2</td>
<td>18.7%</td>
<td>0.08%</td>
<td>19.81%</td>
<td>19.26%</td>
</tr>
<tr>
<td>1.25</td>
<td>19.47%</td>
<td>0.07%</td>
<td>19.13%</td>
<td>19.30%</td>
</tr>
<tr>
<td>1.3</td>
<td>20.35%</td>
<td>0.07%</td>
<td>18.43%</td>
<td>19.39%</td>
</tr>
<tr>
<td>1.35</td>
<td>21.16%</td>
<td>0.06%</td>
<td>17.79%</td>
<td>19.48%</td>
</tr>
</tbody>
</table>

Table 4.50: Experimental Results of the \{Camastra8x8, TrajectoryLength\} Feature Set Using the 300dpi Database

<table>
<thead>
<tr>
<th>$\sigma$</th>
<th>FRR</th>
<th>FAR1</th>
<th>FAR2</th>
<th>AER</th>
</tr>
</thead>
<tbody>
<tr>
<td>14</td>
<td>14.34%</td>
<td>0.11%</td>
<td>23.15%</td>
<td>18.745%</td>
</tr>
<tr>
<td>16</td>
<td>15.21%</td>
<td>0.09%</td>
<td>21.73%</td>
<td>18.47%</td>
</tr>
<tr>
<td>18</td>
<td>16.24%</td>
<td>0.08%</td>
<td>20.31%</td>
<td>18.275%</td>
</tr>
<tr>
<td>20</td>
<td>17.32%</td>
<td>0.07%</td>
<td>18.91%</td>
<td>18.115%</td>
</tr>
<tr>
<td>21.2</td>
<td>18.02%</td>
<td>0.06%</td>
<td>18.08%</td>
<td>18.05%</td>
</tr>
<tr>
<td>22.8</td>
<td>19.09%</td>
<td>0.06%</td>
<td>17.15%</td>
<td>18.12%</td>
</tr>
</tbody>
</table>

Table 4.51: Experimental Results of the \{Variance6+6, TrajectoryLength\} Feature Set Using the 300dpi Database

<table>
<thead>
<tr>
<th>$\sigma$</th>
<th>FRR</th>
<th>FAR1</th>
<th>FAR2</th>
<th>AER</th>
</tr>
</thead>
<tbody>
<tr>
<td>6</td>
<td>17.48%</td>
<td>0.67%</td>
<td>24.72%</td>
<td>21.1%</td>
</tr>
<tr>
<td>8</td>
<td>19.17%</td>
<td>0.56%</td>
<td>22.56%</td>
<td>20.87%</td>
</tr>
<tr>
<td>9.8</td>
<td>20.87%</td>
<td>0.48%</td>
<td>20.72%</td>
<td>20.80%</td>
</tr>
<tr>
<td>10.6</td>
<td>21.74%</td>
<td>0.45%</td>
<td>20.01%</td>
<td>20.88%</td>
</tr>
<tr>
<td>11</td>
<td>22.11%</td>
<td>0.44%</td>
<td>19.64%</td>
<td>20.88%</td>
</tr>
<tr>
<td>12</td>
<td>23.16%</td>
<td>0.4%</td>
<td>18.77%</td>
<td>20.97%</td>
</tr>
<tr>
<td>13</td>
<td>24.29%</td>
<td>0.36%</td>
<td>17.87%</td>
<td>21.08%</td>
</tr>
<tr>
<td>14</td>
<td>25.39%</td>
<td>0.34%</td>
<td>16.98%</td>
<td>21.19%</td>
</tr>
</tbody>
</table>

Table 4.52: Experimental Results of the \{Gradient, TrajectoryLength\} Feature Set Using the 300dpi Database

<table>
<thead>
<tr>
<th>$\sigma$</th>
<th>FRR</th>
<th>FAR1</th>
<th>FAR2</th>
<th>AER</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.43</td>
<td>14.71%</td>
<td>0.02%</td>
<td>15.47%</td>
<td>15.09%</td>
</tr>
<tr>
<td>0.45</td>
<td>15.73%</td>
<td>0.02%</td>
<td>14.29%</td>
<td>15.01%</td>
</tr>
<tr>
<td>0.472</td>
<td>16.87%</td>
<td>0.01%</td>
<td>13.06%</td>
<td>14.965%</td>
</tr>
<tr>
<td>0.476</td>
<td>17.1%</td>
<td>0.01%</td>
<td>12.86%</td>
<td>14.98%</td>
</tr>
<tr>
<td>0.47</td>
<td>16.75%</td>
<td>0.01%</td>
<td>13.14%</td>
<td>14.945%</td>
</tr>
<tr>
<td>0.49</td>
<td>17.9%</td>
<td>0.01%</td>
<td>12.14%</td>
<td>15.02%</td>
</tr>
<tr>
<td>0.50</td>
<td>18.5%</td>
<td>0.01%</td>
<td>11.64%</td>
<td>15.07%</td>
</tr>
</tbody>
</table>

Table 4.53: Experimental Results of the \{Gaussian Grid, TrajectoryLength\} Feature Set Using the 300dpi Database

<table>
<thead>
<tr>
<th>$\sigma$</th>
<th>FRR</th>
<th>FAR1</th>
<th>FAR2</th>
<th>AER</th>
</tr>
</thead>
<tbody>
<tr>
<td>3.75e-2</td>
<td>11.3%</td>
<td>0.03%</td>
<td>17.98%</td>
<td>14.19%</td>
</tr>
<tr>
<td>4.00e-2</td>
<td>12.16%</td>
<td>0.03%</td>
<td>15.85%</td>
<td>14.005%</td>
</tr>
<tr>
<td>4.40e-2</td>
<td>13.76%</td>
<td>0.03%</td>
<td>14.01%</td>
<td>13.885%</td>
</tr>
<tr>
<td>4.50e-2</td>
<td>14.19%</td>
<td>0.02%</td>
<td>13.63%</td>
<td>13.91%</td>
</tr>
<tr>
<td>4.75e-2</td>
<td>15.29%</td>
<td>0.02%</td>
<td>12.59%</td>
<td>13.94%</td>
</tr>
<tr>
<td>5.00e-2</td>
<td>16.53%</td>
<td>0.02%</td>
<td>11.63%</td>
<td>14.08%</td>
</tr>
</tbody>
</table>
Table 4.54: Experimental Results of the \{CurvatureMap, TrajectoryLength\} Feature Set Using the 300dpi Database

<table>
<thead>
<tr>
<th>$\sigma$</th>
<th>FRR</th>
<th>FAR1</th>
<th>FAR2</th>
<th>AER</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.8e-1</td>
<td>17.26%</td>
<td>0.13%</td>
<td>23.15%</td>
<td>20.205%</td>
</tr>
<tr>
<td>1.9e-1</td>
<td>17.97%</td>
<td>0.12%</td>
<td>22.34%</td>
<td>20.155%</td>
</tr>
<tr>
<td>2.0e-1</td>
<td>18.72%</td>
<td>0.11%</td>
<td>21.55%</td>
<td>20.135%</td>
</tr>
<tr>
<td>2.06e-1</td>
<td>19.2%</td>
<td>0.11%</td>
<td>21.02%</td>
<td>20.11%</td>
</tr>
<tr>
<td>2.1e-1</td>
<td>19.48%</td>
<td>0.1%</td>
<td>20.68%</td>
<td><strong>20.08%</strong></td>
</tr>
<tr>
<td>2.14e-1</td>
<td>19.86%</td>
<td>0.1%</td>
<td>20.37%</td>
<td>20.115%</td>
</tr>
<tr>
<td>2.2e-1</td>
<td>20.36%</td>
<td>0.09%</td>
<td>19.87%</td>
<td>20.115%</td>
</tr>
</tbody>
</table>

Table 4.55: Experimental Results of the TrajectoryLength in conjunction with Other Local Feature using the 300dpi Database

<table>
<thead>
<tr>
<th>Local feature</th>
<th>$\sigma$</th>
<th>FRR</th>
<th>FAR1</th>
<th>FAR2</th>
<th>AER</th>
</tr>
</thead>
<tbody>
<tr>
<td>MDF</td>
<td>1.15</td>
<td>17.96%</td>
<td>0.08%</td>
<td>20.51%</td>
<td>19.24%</td>
</tr>
<tr>
<td>Camasta8x8</td>
<td>21.2</td>
<td>18.02%</td>
<td>0.06%</td>
<td>18.08%</td>
<td>18.05%</td>
</tr>
<tr>
<td>Variance6+6</td>
<td>9.8</td>
<td>20.87%</td>
<td>0.48%</td>
<td>20.72%</td>
<td>20.80%</td>
</tr>
<tr>
<td>Gradient</td>
<td>0.47</td>
<td>16.75%</td>
<td>0.01%</td>
<td>14.01%</td>
<td>14.945%</td>
</tr>
<tr>
<td>Gaussian Grid</td>
<td>4.40e-2</td>
<td>13.76%</td>
<td>0.03%</td>
<td>14.01%</td>
<td><strong>13.885%</strong></td>
</tr>
<tr>
<td>Curvature Map</td>
<td>2.1e-1</td>
<td>19.48%</td>
<td>0.1%</td>
<td>20.68%</td>
<td>20.08%</td>
</tr>
</tbody>
</table>

4.3.3 Results Obtained Using the Trajectory Length Feature

Although a similar idea of the Trajectory Length feature was employed in the research of Baltzakis and Papamarkos [14], the discriminating power of this feature was not evaluated in their research. In the present research, the performance of the Trajectory Length was evaluated by comparing the results obtained from local features described in Section 3.3.1 with and without the presence of the Trajectory Length feature. The results of these experiments are displayed in tables from 4.51 to 4.53.

Table 4.55 summarises the experimental results obtained using the Trajectory Length feature.

4.3.4 Results Obtained Using the Moment-based Features

Since Hu’s 1962 foundation work [82], many researchers investigated moment features for various pattern recognition problems including signature verification. Papamarkos and Baltzakis [161] employed the coordinates of the moment centre as global features in their research. Lv et al. [135] represented height to width ratio, incline degree, and extension of signatures using $\mu_{02}$, $\mu_{20}$, and $\mu_{11}$ moments. Siyuan and Srihari [199] investigated the performance of a more sophisticated variant called Zernike-moment.

The moment-based features proposed in this research focus on the rotation-invariant property of the moment centre. The first feature (Moment1) sum the distance between each point of the signature to the moment centre. The second feature, namely Moment3, is enhanced with the sum the inverted distances, which is capable of regularized distant outliers. The experimental results of these features are presented in Section 4.3.4.1 and Section 4.3.4.2, respectively.

4.3.4.1 Results Obtained Using the Moment-based Feature with One Feature Value

The performance of the Moment feature was not evaluated individually as this feature has only one feature value. In the experiments, it was combined with six local features similar to the Trajectory...
Table 4.56: Experimental results of the \{MDF, Moment1\} feature set using the 300dpi database

<table>
<thead>
<tr>
<th>$\sigma$</th>
<th>FRR</th>
<th>FAR1</th>
<th>FAR2</th>
<th>AER</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.05</td>
<td>16.98%</td>
<td>0.1%</td>
<td>21.92%</td>
<td>19.45%</td>
</tr>
<tr>
<td>1.10</td>
<td>17.66%</td>
<td>0.1%</td>
<td>21.15%</td>
<td>19.405%</td>
</tr>
<tr>
<td>1.15</td>
<td>18.38%</td>
<td>0.09%</td>
<td>20.39%</td>
<td>19.385%</td>
</tr>
<tr>
<td>1.22</td>
<td>19.46%</td>
<td>0.08%</td>
<td>19.43%</td>
<td>19.445%</td>
</tr>
<tr>
<td>1.25</td>
<td>19.9%</td>
<td>0.08%</td>
<td>19.01%</td>
<td>19.455%</td>
</tr>
<tr>
<td>1.30</td>
<td>20.72%</td>
<td>0.08%</td>
<td>18.29%</td>
<td>19.505%</td>
</tr>
<tr>
<td>1.35</td>
<td>21.57%</td>
<td>0.07%</td>
<td>17.68%</td>
<td>19.625%</td>
</tr>
</tbody>
</table>

Table 4.57: Experimental results of the \{Camastra8x8, Moment1\} feature set using the 300dpi database

<table>
<thead>
<tr>
<th>$\sigma$</th>
<th>FRR</th>
<th>FAR1</th>
<th>FAR2</th>
<th>AER</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.04</td>
<td>16.77%</td>
<td>0.48%</td>
<td>27.76%</td>
<td>22.265%</td>
</tr>
<tr>
<td>0.1</td>
<td>14.35%</td>
<td>0.65%</td>
<td>29.87%</td>
<td>22.11%</td>
</tr>
<tr>
<td>4</td>
<td>12.76%</td>
<td>0.31%</td>
<td>28.88%</td>
<td>20.82%</td>
</tr>
<tr>
<td>8</td>
<td>13.29%</td>
<td>0.2%</td>
<td>26.6%</td>
<td>19.945%</td>
</tr>
<tr>
<td>19.2</td>
<td>17.27%</td>
<td>0.08%</td>
<td>19.38%</td>
<td>18.325%</td>
</tr>
<tr>
<td>21.6</td>
<td>18.57%</td>
<td>0.07%</td>
<td>17.95%</td>
<td>18.26%</td>
</tr>
<tr>
<td>23.2</td>
<td>19.58%</td>
<td>0.06%</td>
<td>17.08%</td>
<td>18.33%</td>
</tr>
</tbody>
</table>

Table 4.58: Experimental Results of the \{Variance6+6, Moment1\} Feature Set Using the 300dpi Database

<table>
<thead>
<tr>
<th>$\sigma$</th>
<th>FRR</th>
<th>FAR1</th>
<th>FAR2</th>
<th>AER</th>
</tr>
</thead>
<tbody>
<tr>
<td>6e-2</td>
<td>23.82%</td>
<td>1.15%</td>
<td>27.26%</td>
<td>25.54%</td>
</tr>
<tr>
<td>8e-2</td>
<td>21.19%</td>
<td>1.24%</td>
<td>28.56%</td>
<td>24.875%</td>
</tr>
<tr>
<td>2</td>
<td>16.2%</td>
<td>1.07%</td>
<td>28.33%</td>
<td>22.265%</td>
</tr>
<tr>
<td>4</td>
<td>17.5%</td>
<td>0.86%</td>
<td>25.93%</td>
<td>21.715%</td>
</tr>
<tr>
<td>5</td>
<td>18.19%</td>
<td>0.78%</td>
<td>24.89%</td>
<td>21.54%</td>
</tr>
<tr>
<td>7</td>
<td>19.68%</td>
<td>0.66%</td>
<td>22.96%</td>
<td>21.32%</td>
</tr>
<tr>
<td>8</td>
<td>20.5%</td>
<td>0.61%</td>
<td>22.04%</td>
<td>21.27%</td>
</tr>
<tr>
<td>8.8</td>
<td>21.16%</td>
<td>0.58%</td>
<td>21.27%</td>
<td>21.215%</td>
</tr>
<tr>
<td>9.8</td>
<td>22.13%</td>
<td>0.53%</td>
<td>20.46%</td>
<td>21.295%</td>
</tr>
</tbody>
</table>

Table 4.59: Experimental results of the \{Gradient, Moment1\} Feature Set Using the 300dpi Database

<table>
<thead>
<tr>
<th>$\sigma$</th>
<th>FRR</th>
<th>FAR1</th>
<th>FAR2</th>
<th>AER</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.42</td>
<td>14.26%</td>
<td>0.02%</td>
<td>16.15%</td>
<td>15.205%</td>
</tr>
<tr>
<td>0.44</td>
<td>15.22%</td>
<td>0.02%</td>
<td>14.98%</td>
<td>15.1%</td>
</tr>
<tr>
<td>0.448</td>
<td>15.66%</td>
<td>0.02%</td>
<td>14.51%</td>
<td>15.085%</td>
</tr>
<tr>
<td>0.452</td>
<td>15.88%</td>
<td>0.02%</td>
<td>14.28%</td>
<td>15.08%</td>
</tr>
<tr>
<td>0.456</td>
<td>16.1%</td>
<td>0.02%</td>
<td>14.09%</td>
<td>15.095%</td>
</tr>
<tr>
<td>0.468</td>
<td>16.7%</td>
<td>0.02%</td>
<td>13.41%</td>
<td>15.055%</td>
</tr>
<tr>
<td>0.48</td>
<td>17.37%</td>
<td>0.02%</td>
<td>12.78%</td>
<td>15.075%</td>
</tr>
<tr>
<td>0.50</td>
<td>18.59%</td>
<td>0.02%</td>
<td>11.75%</td>
<td>15.17%</td>
</tr>
</tbody>
</table>
Table 4.60: Experimental Results of the {GaussianGrid, Moment1} Feature Set Using the 300dpi Database

<table>
<thead>
<tr>
<th>( \sigma )</th>
<th>FRR</th>
<th>FAR1</th>
<th>FAR2</th>
<th>AER</th>
</tr>
</thead>
<tbody>
<tr>
<td>4.0e-2</td>
<td>12.17%</td>
<td>0.03%</td>
<td>15.8%</td>
<td>13.985%</td>
</tr>
<tr>
<td>4.28e-2</td>
<td>13.26%</td>
<td>0.03%</td>
<td>14.52%</td>
<td>13.895%</td>
</tr>
<tr>
<td>4.32e-2</td>
<td>13.44%</td>
<td>0.03%</td>
<td>14.33%</td>
<td>13.885%</td>
</tr>
<tr>
<td>4.4e-2</td>
<td>13.75%</td>
<td>0.02%</td>
<td>14.01%</td>
<td>13.885%</td>
</tr>
<tr>
<td>4.52e-2</td>
<td>14.24%</td>
<td>0.02%</td>
<td>13.53%</td>
<td>13.885%</td>
</tr>
<tr>
<td>4.6e-2</td>
<td>14.61%</td>
<td>0.02%</td>
<td>13.24%</td>
<td>13.925%</td>
</tr>
<tr>
<td>4.8e-2</td>
<td>15.52%</td>
<td>0.02%</td>
<td>12.43%</td>
<td>13.975%</td>
</tr>
<tr>
<td>5.0e-2</td>
<td>16.58%</td>
<td>0.02%</td>
<td>11.65%</td>
<td>14.115%</td>
</tr>
</tbody>
</table>

Table 4.61: Experimental Results of the {CurvatureMap, Moment1} Feature Set Using the 300dpi Database

<table>
<thead>
<tr>
<th>( \sigma )</th>
<th>FRR</th>
<th>FAR1</th>
<th>FAR2</th>
<th>AER</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.8e-1</td>
<td>17.5%</td>
<td>0.13%</td>
<td>23.04%</td>
<td>20.27%</td>
</tr>
<tr>
<td>1.9e-1</td>
<td>18.23%</td>
<td>0.12%</td>
<td>22.25%</td>
<td>20.24%</td>
</tr>
<tr>
<td>2.0e-1</td>
<td>19%</td>
<td>0.11%</td>
<td>21.44%</td>
<td>20.22%</td>
</tr>
<tr>
<td>2.06e-1</td>
<td>19.48%</td>
<td>0.11%</td>
<td>20.95%</td>
<td>20.215%</td>
</tr>
<tr>
<td>2.12e-1</td>
<td>19.95%</td>
<td>0.1%</td>
<td>20.43%</td>
<td><strong>20.19%</strong></td>
</tr>
<tr>
<td>2.18e-1</td>
<td>20.46%</td>
<td>0.1%</td>
<td>19.96%</td>
<td>20.21%</td>
</tr>
<tr>
<td>2.2e-1</td>
<td>20.65%</td>
<td>0.09%</td>
<td>19.8%</td>
<td>20.225%</td>
</tr>
</tbody>
</table>

Length feature. The results of these combinations are presented in tables from 4.56 to 4.61.

### 4.3.4.2 Results Obtained Using the Moment-based Feature with Two Feature Values

Similar to the previous section, this section presents the results of the 2-value Moment-based feature in conjunction with other local features. Tables from 4.62 to 4.67 detail the results obtained. It is not expected that this variant of moment-based feature clearly outperform the previous one especially in the combination with high profile local feature (e.g. Gaussian Grid or Gradient). However, better performance should be clearly evident in experiments with low profile local features such as the Variance6+6 or Camastra features.

### 4.4 Results of Feature Fusion

Finding the combination that produces the best result from a pool of available features is an open problem in pattern recognition. Most of the investigations performed in the field of signature

Table 4.62: Experimental results of the {e-MDF, Moment3} feature set using the 300dpi database

<table>
<thead>
<tr>
<th>( \sigma )</th>
<th>FRR</th>
<th>FAR1</th>
<th>FAR2</th>
<th>AER</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.8</td>
<td>13.53%</td>
<td>0.15%</td>
<td>26.09%</td>
<td>19.81%</td>
</tr>
<tr>
<td>1.05</td>
<td>16.79%</td>
<td>0.1%</td>
<td>21.91%</td>
<td>19.35%</td>
</tr>
<tr>
<td>1.13</td>
<td>17.96%</td>
<td>0.09%</td>
<td>20.67%</td>
<td><strong>19.315%</strong></td>
</tr>
<tr>
<td>1.18</td>
<td>18.73%</td>
<td>0.09%</td>
<td>19.96%</td>
<td>19.345%</td>
</tr>
<tr>
<td>1.25</td>
<td>19.78%</td>
<td>0.08%</td>
<td>18.97%</td>
<td>19.375%</td>
</tr>
<tr>
<td>1.3</td>
<td>20.61%</td>
<td>0.07%</td>
<td>18.32%</td>
<td>19.465%</td>
</tr>
<tr>
<td>1.35</td>
<td>21.46%</td>
<td>0.07%</td>
<td>17.68%</td>
<td>19.57%</td>
</tr>
<tr>
<td>1.6</td>
<td>25.67%</td>
<td>0.05%</td>
<td>14.53%</td>
<td>20.1%</td>
</tr>
</tbody>
</table>
Table 4.63: Experimental results of the \{Camastra8x8, Moment3\} feature set using the 300dpi database

<table>
<thead>
<tr>
<th>(\sigma)</th>
<th>FRR</th>
<th>FAR1</th>
<th>FAR2</th>
<th>AER</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.04</td>
<td>15.16%</td>
<td>0.48%</td>
<td>28.46%</td>
<td>21.81%</td>
</tr>
<tr>
<td>0.4</td>
<td>12.67%</td>
<td>0.62%</td>
<td>30.66%</td>
<td>21.665%</td>
</tr>
<tr>
<td>4.0</td>
<td>11.5%</td>
<td>0.3%</td>
<td>29.37%</td>
<td>20.435%</td>
</tr>
<tr>
<td>12.0</td>
<td>12.98%</td>
<td>0.12%</td>
<td>24.28%</td>
<td>18.63%</td>
</tr>
<tr>
<td>16.0</td>
<td>14.44%</td>
<td>0.06%</td>
<td>21.48%</td>
<td>17.96%</td>
</tr>
<tr>
<td>18.0</td>
<td>15.3%</td>
<td>0.07%</td>
<td>20.13%</td>
<td>17.715%</td>
</tr>
<tr>
<td>20.0</td>
<td>16.39%</td>
<td>0.06%</td>
<td>18.81%</td>
<td>17.6%</td>
</tr>
<tr>
<td>22.0</td>
<td>17.48%</td>
<td>0.06%</td>
<td>17.62%</td>
<td>17.55%</td>
</tr>
<tr>
<td>24.0</td>
<td>18.75%</td>
<td>0.05%</td>
<td>16.46%</td>
<td>17.605%</td>
</tr>
<tr>
<td>25.0</td>
<td>19.42%</td>
<td>0.05%</td>
<td>15.86%</td>
<td>17.64%</td>
</tr>
</tbody>
</table>

Table 4.64: Experimental Results of the \{Variance6+6, Moment3\} Feature Set Using the 300dpi Database

<table>
<thead>
<tr>
<th>(\sigma)</th>
<th>FRR</th>
<th>FAR1</th>
<th>FAR2</th>
<th>AER</th>
</tr>
</thead>
<tbody>
<tr>
<td>2.0</td>
<td>14.61%</td>
<td>0.99%</td>
<td>28.96%</td>
<td>21.785%</td>
</tr>
<tr>
<td>4.0</td>
<td>15.93%</td>
<td>0.77%</td>
<td>26.31%</td>
<td>21.12%</td>
</tr>
<tr>
<td>6.0</td>
<td>17.45%</td>
<td>0.63%</td>
<td>24.04%</td>
<td>20.745%</td>
</tr>
<tr>
<td>8.2</td>
<td>19.41%</td>
<td>0.53%</td>
<td>21.86%</td>
<td>20.635%</td>
</tr>
<tr>
<td>8.8</td>
<td>19.98%</td>
<td>0.5%</td>
<td>21.32%</td>
<td>20.65%</td>
</tr>
<tr>
<td>9.2</td>
<td>20.35%</td>
<td>0.48%</td>
<td>20.95%</td>
<td>20.65%</td>
</tr>
<tr>
<td>9.6</td>
<td>20.78%</td>
<td>0.46%</td>
<td>20.6%</td>
<td>20.69%</td>
</tr>
<tr>
<td>10.0</td>
<td>21.19%</td>
<td>0.45%</td>
<td>20.26%</td>
<td>20.725%</td>
</tr>
</tbody>
</table>

Table 4.65: Experimental results of the \{Gradient, Moment3\} Feature Set Using the 300dpi Database

<table>
<thead>
<tr>
<th>(\sigma)</th>
<th>FRR</th>
<th>FAR1</th>
<th>FAR2</th>
<th>AER</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.42</td>
<td>14.27%</td>
<td>0.02%</td>
<td>16.11%</td>
<td>15.19%</td>
</tr>
<tr>
<td>0.43</td>
<td>14.73%</td>
<td>0.02%</td>
<td>15.52%</td>
<td>15.125%</td>
</tr>
<tr>
<td>0.44</td>
<td>15.19%</td>
<td>0.02%</td>
<td>14.93%</td>
<td>15.06%</td>
</tr>
<tr>
<td>0.45</td>
<td>15.73%</td>
<td>0.02%</td>
<td>14.38%</td>
<td>15.055%</td>
</tr>
<tr>
<td>0.46</td>
<td>16.21%</td>
<td>0.02%</td>
<td>13.8%</td>
<td>15.005%</td>
</tr>
<tr>
<td>0.47</td>
<td>16.75%</td>
<td>0.01%</td>
<td>13.22%</td>
<td>14.985%</td>
</tr>
<tr>
<td>0.48</td>
<td>17.32%</td>
<td>0.01%</td>
<td>12.67%</td>
<td>14.995%</td>
</tr>
</tbody>
</table>

Table 4.66: Experimental Results of the \{GaussianGrid, Moment3\} Feature Set Using the 300dpi Database

<table>
<thead>
<tr>
<th>(\sigma)</th>
<th>FRR</th>
<th>FAR1</th>
<th>FAR2</th>
<th>AER</th>
</tr>
</thead>
<tbody>
<tr>
<td>4.0e-2</td>
<td>12.22%</td>
<td>0.03%</td>
<td>15.85%</td>
<td>14.035%</td>
</tr>
<tr>
<td>4.3e-2</td>
<td>13.33%</td>
<td>0.03%</td>
<td>14.49%</td>
<td>13.91%</td>
</tr>
<tr>
<td>4.4e-2</td>
<td>13.71%</td>
<td>0.03%</td>
<td>14.06%</td>
<td>13.885%</td>
</tr>
<tr>
<td>4.5e-2</td>
<td>14.15%</td>
<td>0.03%</td>
<td>13.63%</td>
<td>13.89%</td>
</tr>
<tr>
<td>4.6e-2</td>
<td>14.6%</td>
<td>0.02%</td>
<td>13.17%</td>
<td>13.885%</td>
</tr>
<tr>
<td>4.7e-2</td>
<td>15.04%</td>
<td>0.02%</td>
<td>12.78%</td>
<td>13.91%</td>
</tr>
<tr>
<td>4.8e-2</td>
<td>15.54%</td>
<td>0.02%</td>
<td>12.39%</td>
<td>13.965%</td>
</tr>
<tr>
<td>5.0e-2</td>
<td>16.53%</td>
<td>0.02%</td>
<td>11.66%</td>
<td>14.095%</td>
</tr>
</tbody>
</table>
Table 4.67: Experimental Results of the \{CurvatureMap, Moment3\} Feature Set Using the 300dpi Database

<table>
<thead>
<tr>
<th>σ</th>
<th>FRR</th>
<th>FAR1</th>
<th>FAR2</th>
<th>AER</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.9e-1</td>
<td>18.01%</td>
<td>0.13%</td>
<td>22.35%</td>
<td>20.18%</td>
</tr>
<tr>
<td>2.02e-1</td>
<td>18.93%</td>
<td>0.12%</td>
<td>21.35%</td>
<td>20.14%</td>
</tr>
<tr>
<td>2.08e-1</td>
<td>19.43%</td>
<td>0.11%</td>
<td>20.84%</td>
<td>20.135%</td>
</tr>
<tr>
<td>2.1e-1</td>
<td>19.59%</td>
<td>0.11%</td>
<td>20.68%</td>
<td>20.135%</td>
</tr>
<tr>
<td>2.14e-1</td>
<td>19.96%</td>
<td>0.1%</td>
<td>20.41%</td>
<td>20.185%</td>
</tr>
<tr>
<td>2.2e-1</td>
<td>20.56%</td>
<td>0.1%</td>
<td>19.88%</td>
<td>20.22%</td>
</tr>
<tr>
<td>2.3e-1</td>
<td>21.51%</td>
<td>0.09%</td>
<td>19.03%</td>
<td>20.27%</td>
</tr>
<tr>
<td>2.4e-1</td>
<td>22.49%</td>
<td>0.08%</td>
<td>18.22%</td>
<td>20.355%</td>
</tr>
</tbody>
</table>

Table 4.68: Experimental Results of the \{MDF, E5, R2, Trajectory Length, Moment\} Feature Set Using the 300dpi Database

<table>
<thead>
<tr>
<th>σ</th>
<th>FRR</th>
<th>FAR1</th>
<th>FAR2</th>
<th>AER</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.9</td>
<td>13.12%</td>
<td>0.09%</td>
<td>23.99%</td>
<td>18.56%</td>
</tr>
<tr>
<td>1.0</td>
<td>14.53%</td>
<td>0.07%</td>
<td>22.15%</td>
<td>18.34%</td>
</tr>
<tr>
<td>1.1</td>
<td>15.95%</td>
<td>0.07%</td>
<td>20.36%</td>
<td>18.155%</td>
</tr>
<tr>
<td>1.12</td>
<td>16.28%</td>
<td>0.06%</td>
<td>20.05%</td>
<td>18.17%</td>
</tr>
<tr>
<td>1.16</td>
<td>16.93%</td>
<td>0.06%</td>
<td>19.34%</td>
<td>18.14%</td>
</tr>
<tr>
<td>1.18</td>
<td>17.25%</td>
<td>0.06%</td>
<td>19.02%</td>
<td>18.14%</td>
</tr>
<tr>
<td>1.2</td>
<td>17.62%</td>
<td>0.06%</td>
<td>18.71%</td>
<td>18.17%</td>
</tr>
<tr>
<td>1.3</td>
<td>19.31%</td>
<td>0.05%</td>
<td>17.21%</td>
<td>18.26%</td>
</tr>
</tbody>
</table>

verification involve small dimensional features [64, 186]. Therefore an exhaustive search for the top performing feature set using the techniques investigated is not within the scope of this research. The choice of feature combinations under investigation in this section were performed heuristically.

Tables 4.69, 4.68, and 4.70 respectively present the results for the Variance6+6, MDF, and Camastra4x4 when being combined with a set of global features including the Moment Feature, E5, R2, and Trajectory Length. The dimensions of the feature vector upon undertaking these feature fusions are 32, 127, and 40 respectively.

Despite the relatively disappointing results when being used individually, the Curvature Map feature slightly improves the performance of the Gaussian Grid feature. Table 4.71 presents the results of this experiment. As may be seen, an AER of 13.54% was obtained. The total dimension of this feature combination is 1152.

This concludes the chapter for experimental results. The following chapter interprets the numerous results presented. The significance of the results will also be discussed and a comparison

Table 4.69: Experimental Results of the \{Variance6+6, E5, R2, Trajectory Length, Moment3\} Feature Set Using the 300dpi database

<table>
<thead>
<tr>
<th>σ</th>
<th>FRR</th>
<th>FAR1</th>
<th>FAR2</th>
<th>AER</th>
</tr>
</thead>
<tbody>
<tr>
<td>7.0</td>
<td>13.43%</td>
<td>0.22%</td>
<td>20.48%</td>
<td>16.955%</td>
</tr>
<tr>
<td>7.7</td>
<td>14.68%</td>
<td>0.2%</td>
<td>19.02%</td>
<td>16.85%</td>
</tr>
<tr>
<td>8.0</td>
<td>15.16%</td>
<td>0.19%</td>
<td>18.43%</td>
<td>16.80%</td>
</tr>
<tr>
<td>8.4</td>
<td>15.97%</td>
<td>0.17%</td>
<td>17.71%</td>
<td>16.84%</td>
</tr>
<tr>
<td>8.6</td>
<td>16.38%</td>
<td>0.16%</td>
<td>17.37%</td>
<td>16.875%</td>
</tr>
<tr>
<td>8.8</td>
<td>16.83%</td>
<td>0.16%</td>
<td>17.02%</td>
<td>16.925%</td>
</tr>
<tr>
<td>9.0</td>
<td>17.23%</td>
<td>0.15%</td>
<td>16.66%</td>
<td>16.945%</td>
</tr>
<tr>
<td>10.0</td>
<td>19.18%</td>
<td>0.13%</td>
<td>14.98%</td>
<td>17.08%</td>
</tr>
</tbody>
</table>
### Table 4.70: Experimental Results of the \{Camstra4x4, E5, R2, Trajectory Length, Moment3\} Feature Set Using the 300dpi database

<table>
<thead>
<tr>
<th>$\sigma$</th>
<th>FRR</th>
<th>FAR1</th>
<th>FAR2</th>
<th>AER</th>
</tr>
</thead>
<tbody>
<tr>
<td>20</td>
<td>13.39%</td>
<td>0.15%</td>
<td>19.49%</td>
<td>16.44%</td>
</tr>
<tr>
<td>22</td>
<td>14.75%</td>
<td>0.12%</td>
<td>18.02%</td>
<td>16.385%</td>
</tr>
<tr>
<td>24</td>
<td>16.22%</td>
<td>0.11%</td>
<td>16.59%</td>
<td>16.405%</td>
</tr>
<tr>
<td>26</td>
<td>17.61%</td>
<td>0.09%</td>
<td>15.15%</td>
<td><strong>16.38%</strong></td>
</tr>
<tr>
<td>28</td>
<td>19.15%</td>
<td>0.08%</td>
<td>13.85%</td>
<td>16.5%</td>
</tr>
<tr>
<td>30</td>
<td>20.87%</td>
<td>0.07%</td>
<td>12.6%</td>
<td>16.735%</td>
</tr>
<tr>
<td>32</td>
<td>22.69%</td>
<td>0.06%</td>
<td>11.39%</td>
<td>17.04%</td>
</tr>
<tr>
<td>34</td>
<td>24.6%</td>
<td>0.05%</td>
<td>10.31%</td>
<td>17.455%</td>
</tr>
</tbody>
</table>

### Table 4.71: Experimental Results of the \{GaussianGrid, CurvatureMap\} Feature Set using the 300dpi Database

<table>
<thead>
<tr>
<th>$\sigma$</th>
<th>FRR</th>
<th>FAR1</th>
<th>FAR2</th>
<th>AER</th>
</tr>
</thead>
<tbody>
<tr>
<td>3.92e-2</td>
<td>13.61%</td>
<td>0.02%</td>
<td>13.46%</td>
<td><strong>13.535%</strong></td>
</tr>
<tr>
<td>4.00e-2</td>
<td>14.12%</td>
<td>0.01%</td>
<td>13.05%</td>
<td>13.585%</td>
</tr>
<tr>
<td>4.06e-2</td>
<td>14.5%</td>
<td>0.01%</td>
<td>12.72%</td>
<td>13.61%</td>
</tr>
<tr>
<td>4.10e-2</td>
<td>14.74%</td>
<td>0.01%</td>
<td>12.47%</td>
<td>13.605%</td>
</tr>
<tr>
<td>4.20e-2</td>
<td>15.35%</td>
<td>0.01%</td>
<td>11.96%</td>
<td>13.655%</td>
</tr>
<tr>
<td>4.30e-2</td>
<td>16.01%</td>
<td>0.01%</td>
<td>11.44%</td>
<td>13.725%</td>
</tr>
<tr>
<td>4.40e-2</td>
<td>16.72%</td>
<td>0.01%</td>
<td>10.93%</td>
<td>13.825%</td>
</tr>
<tr>
<td>4.50e-2</td>
<td>17.43%</td>
<td>0.01%</td>
<td>10.53%</td>
<td>13.98%</td>
</tr>
</tbody>
</table>

to other results in the literature will be provided.
Chapter 5

ANALYSIS AND COMPARISON OF RESULTS

From the previous chapter, it can be seen that a large number of experiments have been conducted in this research. The primary purpose of those experiments were to investigate the performance of state-of-art feature extraction techniques including MDF, Camastra, and the Gradient Feature for signature verification. Particularly, various aspects of the MDF were examined in detail in order to improve the performance of this feature. In addition, the experiments provide an insight into the performance of the newly proposed feature extraction techniques and feature sets. This chapter analyses and interprets the implication of the results as well as discusses their significance. The comparison of the results obtained to those of other authors will also be presented in this chapter.

This chapter is organized into five main sections. Firstly, Section 5.1 analyses the results and performance of the newly proposed trajectory recovery framework. Secondly, the analysis of results obtained using local features will be presented in Section 5.2. This is followed by Section 5.3 which discusses the results of global feature extraction. Section 5.4 reviews the notable results of feature fusion. Lastly and most importantly, comparisons to the results of other researchers in the field are presented in Section 5.5.

5.1 Analysis of Trajectory Recovery Results

5.1.1 Analysis of Results of the Curvature Maxima Locator Algorithm

Distinct from the simulated data employed in other research [107, 181], the US-SIGBASE corpus employed in this thesis is a real-world collection of static signatures. No contour smoothing process had been performed and consequently, the contours were not as smooth as those of the simulated static signatures. Nevertheless, anomalies often occur at stroke intersections as a result of surface tension whenever wet ink was used to produce the signature. This makes the correct localisation of curvature maxima a more difficult task.

To compensate for the deformation, up to two pixels were gradually removed from the connectors of the contours in the calculation of the bridging score and only the highest score was retained. Unfortunately, this strategy can only be applied for contour segments longer than a certain threshold.

Based on visual inspections, the results indicated that the proposed curvature maxima locator
algorithm (CMLA) had not been very successful on the US-SIGBASE database. The crucial maxima were located completely in only 37% of the tested signatures. The percentage of signatures with 3 or more missing maxima was as high as 22%. The majority of missing maxima were caused by the insufficient size of the supporting area. The missing maxima in figures 4.1.4k and 4.1.4n are typical examples.

5.1.2 Analysis of Results of the Proposed Trajectory Recovery Framework

The performance of the proposed trajectory recovery framework heavily relies on correct maxima points located by the CMLA. A missing local curvature maximum point often results in two erroneous matching pairs. This phenomenon is clearly illustrated in Figure 4.1.4k with three incorrectly matched pairs of contour segments. In total, the proposed framework successfully analysed 6 signatures (09, 13, 16, 36, 45, 46).

5.2 Analysis of Results Obtained Using Local Features

5.2.1 Analysis of Results Obtained Using the e-MDF Feature

A large number of experiments have been conducted using the MDF feature and its variants in the current research in an attempt to improve its performance for the signature verification problem.

As expected, the experimental results, which were presented in sections 4.2.1.1 and 4.2.1.2, showed the effectiveness of the newly proposed grouping technique. In the experiments using the 75dpi database, the MDF produced the best AER of 18.23% whilst this figure for the e-MDF was 17.75%. Improvements were also observed in combination with other features. Figure 5.2.1 depicts the results of these experiments.

The experimental results also indicated that the resolution of the input images have a significant impact on the performance of the MDF. This is due to the deformation of the contours when the resolution varies. Specifically, the MDF produced better results using the lower resolution database of 75dpi. A substantial difference in AER of up to 1.86% was observed in the experiments using the enhanced MDF in conjunction with the R2 feature. The AERs of the 75dpi and 300dpi experiments were 17.39% and 19.25%, respectively. At the lower resolution of 75dpi, the width of many strokes was only 1 or 2 pixels and there was no blank gap between the sides of those strokes. Therefore

Figure 5.2.1: The performance of the MDF-based feature sets on the 75dpi database
only one transition was recorded instead of two as in thicker strokes. This probably enabled the MDF to capture more information from inner transitions. However, this wasn’t the case as the experimental results indicated. Figure 5.2.2 summarises the performance of several MDF-based feature sets at the resolution of 75dpi and 300dpi.

Figure 5.2.3 presents the best average error rates of the enhanced MDF and the MDF variant with only the first 2 layers of transitions (MDF_L12) in some selective feature combinations. It can be seen that the performance of MDF_L12 is similar to that of the recommended configuration of the MDF. Depending on which feature combination, the accuracies of the MDF with only the first two layers of transition can be up to 0.2% lower than that of the original configuration. One possible explanation for the difference is that the computation of the 3rd-transition feature values are less stable (due to the accumulation of instability of outer transitions) compared to the 1st and 2nd layers in signatures, and provides little contribution to the overall performance of the MDF.
CHAPTER 5. ANALYSIS AND COMPARISON OF RESULTS

Two additional experiments were performed using the 300dpi database to check this hypothesis. In these experiments, data values from the 1\textsuperscript{st} and 3\textsuperscript{rd} were removed to see if the experimental results without the 1\textsuperscript{st} layer of transition is substantially lower than those without the 3\textsuperscript{rd} layer. Figure 5.2.4 depicts the ROC curves using the results previously reported in tables 4.6 and 4.7 for these two experiments.

The proposed hypothesis was supported by the results obtained. The best average error rate of the MDF with the first layer of transition removed was about 0.43% higher than the MDF without the 3\textsuperscript{rd} layer of transition.

Nevertheless, it must be noted that the MDF Feature is computed using pattern contours which consist of two parallel edges. Therefore there is not much difference between the values on the first and second layers. The best AER of 20.02% was obtained from the experiments of MDF without layer 2 feature values, as presented in Table 4.8, which was almost the same as the value obtained using the MDF_L23 Feature (19.99%).

In another experiment, both feature values (and 3\textsuperscript{rd} layers) were removed from the MDF. The results obtained, which are reported in Table 4.9, indicate that although the discrimination power of feature values in the 1\textsuperscript{st} and 2\textsuperscript{nd} layers are similar, their cooperation produced better results. The best AER obtained of 19.95% was 0.39% higher than that obtained with the MDF_L12 (19.56%). Results from the above experiments indicate that the feature values in every layer of transitions contribute to the performance of the MDF for signature verification. However, the discrimination power of feature values in the 1\textsuperscript{st} and 2\textsuperscript{nd} transition layers are similar and higher than their counterparts in the 3\textsuperscript{rd} layer.

Whilst the inclusion of the 3\textsuperscript{rd} layer of transition improved the performance of the MDF by 0.12% (from 19.50% AER down to 19.47% AER) in the experiments using the 300 dpi database, increasing the number of groups appeared to be a more effective way to improve the performance of the MDF Feature. As the number of groups increased to 15, the best AER obtained was 18.85%. In this configuration, the dimensions of the feature vector was $15 \times 3 \times 4 \times 2 = 360$, which is three times larger than the original configuration. Figure 5.2.5 illustrates the impact of the number of groups on the best AER.

![Figure 5.2.5: FAR, FRR\textsuperscript{2}, and AER of the MDF with different numbers of groups](image)
5.2.2 Analysis of Results Obtained Using the Camastra Feature

The results presented in Section 4.2.2 have shown that the Camastra Feature produced better cursive character recognition results compared to the MDF Feature. However, as signatures are more complex patterns, the simplicity of the Camastra feature appeared to be less appropriate for signature verification compared with the MDF feature.

In its original grid settings, which were employed for the character recognition problem, the Camastra feature produced a relatively high AER of 20.33%. This figure is 2.10% higher than the AER obtained with the original MDF Feature using the 75dpi database and 0.86% higher than the AER obtained using the enhanced MDF for the 300dpi database. The performance of the Camastra Feature peaked at 19.35% AER with the grid dimensions of $8 \times 8$ (Camastra8x8). This result is comparable to the AER obtained using the enhanced MDF for the 300dpi database which is 19.47%. Interestingly, the dimensions of the Camastra8x8 feature vector approximates that of the MDF (128 and 120). However, the performance of the MDF could be slightly improved by increasing the number of groups. The 30-group variant of the MDF feature produced an AER of 18.73% for the 300 dpi database (see Table 4.5).

5.2.3 Analysis of Results Obtained Using the Gradient Feature

Table 5.1: Experimental Results of the Gradient Feature

<table>
<thead>
<tr>
<th>Feature</th>
<th>Classifier</th>
<th>FAR</th>
<th>FRR</th>
<th>AER</th>
</tr>
</thead>
<tbody>
<tr>
<td>Gradient</td>
<td>SVM</td>
<td>16.54%</td>
<td>13.51%</td>
<td>15.03%</td>
</tr>
<tr>
<td>Gradient</td>
<td>$d^2_M(x)$</td>
<td>14.80%</td>
<td>18.63%</td>
<td>16.52%</td>
</tr>
</tbody>
</table>

In the experiments using the gradient feature, the squared Mahalanobis distance classifier produced an AER of 16.52% and an EER of 16.77%. Taking into account the computation required, these results are very encouraging in comparison with SVMs. When the SVMs were employed, the gradient feature produced the lowest AER of 15.03% and the EER of 15.11%. The FAR and FRR were 16.54% and 13.51% respectively where the best AER was obtained. Table 5.1 summarises the key results of the Gradient feature.

Figure 5.2.6: The ROC curves of the MDF and gradient Feature obtained using SVMs and the squared Mahalanobis distance
Table 5.2: Performance comparison for the Variance6+6 with global features

<table>
<thead>
<tr>
<th>Feature Set</th>
<th>Database Size</th>
<th>Size</th>
<th>AER</th>
</tr>
</thead>
<tbody>
<tr>
<td>Variance6+6 with global features</td>
<td>300dpi</td>
<td>32</td>
<td>17.02%</td>
</tr>
<tr>
<td>Variance12x12 with global features</td>
<td>300dpi</td>
<td>56</td>
<td>17.69%</td>
</tr>
<tr>
<td>Variance6+6 with global features (Moment3)</td>
<td>300dpi</td>
<td>33</td>
<td>16.80%</td>
</tr>
</tbody>
</table>

It can be concluded that the Gradient feature outperforms the MDF and Camastra features whose AER were 17.75% and 19.35% respectively. This could be partly explained by the gradient feature’s substantially larger feature vector of 576 dimensions, which is nearly five times the size of MDF. The massive size of the Gradient Feature allows it to describe signatures more precisely. Nevertheless, the blurring process also plays a crucial role of preserving information which stabilises the feature values extracted against variation. Figure 5.2.6 presents the ROC curves of the Gradient feature obtained using SVMs and the Squared Mahalanobis Distance classifiers.

5.2.4 Analysis of Results Obtained Using the Variance Feature

The results reported in Table 4.16 indicate that the performance of the Variance Feature itself is not as good as other local features investigated in this research. In the 6-segment setting, the AER is 22.20%. The performance is slightly improved and peaked at 21.85% AER in the 12-segment setting. These results are worse than other local features but they are not unexpected. The Variance Feature only consists of a small number of feature values which are comprised of simple statistic information. In the 6-group setting, the dimension of the feature vector is only 24.

Despite the low accuracies, the performance of the Variance feature was significantly improved when additional global features were employed. Results from Table 4.69 showed that an AER of 17.02% was obtained with the \{Variance6+6, R2, E5, Trajectory Length, Moment\} feature set. That is 5.18% improvement with the assistance of 4 additional global features with 8 feature values. In total, the number of the feature values of this feature set is only 32, which is a quarter of the Camastra8x8 and nearly a quarter of the MDF. Table 5.2 compares the dimensions and the AER of these feature sets.

In the experiments with the 12-segment variant of the Variance feature, the global features also produced a substantial improvement of 4.17%, from 21.86% AER down to 17.69% AER. Surprisingly, this result is 0.67% higher than that obtained when using the 6-segment variant. When being employed individually, the AER of the 12-segment variant is 0.76% better than the AER of the 6-segment variant. It is suggested that the influence of the global features is larger when the dimension of the local feature vector is smaller.

5.2.5 Analysis of Results Obtained Using the Gaussian Grid Feature

Table 5.3: Experimental Results Obtained using the Gaussian Grid Feature

<table>
<thead>
<tr>
<th>Grid Size</th>
<th>Filter</th>
<th>FRR</th>
<th>FAR$^1$</th>
<th>FAR$^2$</th>
<th>AER</th>
</tr>
</thead>
<tbody>
<tr>
<td>9 × 9$\sigma=1.2$</td>
<td>Yes</td>
<td>14.37%</td>
<td>0.04%</td>
<td>14.42%</td>
<td>14.40%</td>
</tr>
<tr>
<td>12 × 12$\sigma=1.2$</td>
<td>Yes</td>
<td>14.18%</td>
<td>0.02%</td>
<td>13.68%</td>
<td>13.93%</td>
</tr>
<tr>
<td>9 × 9$\sigma=1.2$</td>
<td>No</td>
<td>17.97%</td>
<td>0.04%</td>
<td>18.08%</td>
<td>18.03%</td>
</tr>
<tr>
<td>12 × 12$\sigma=1.2$</td>
<td>No</td>
<td>19.48%</td>
<td>0.06%</td>
<td>19.84%</td>
<td>19.66%</td>
</tr>
</tbody>
</table>
The performance of the Gaussian Grid Feature is very encouraging. As summarized in Table 5.3, this feature produced an AER of 13.93% when the 12 × 12 grid configuration was employed. This result is substantially better than those produced by other local features investigated in this research. Even with a smaller dimension configuration (9 × 9 × 6 = 486) the Gaussian Grid Feature produces a better AER of 14.40%. The ROC curves of these experiments are depicted in Figure 5.2.8. Apart from that, the FAR rate for random forgeries (FAR2) was also kept at an extremely low rate of 0.02% (i.e., 56 out of 30 × 160 × 59 = 283200 tests). It is believed that this error rate could be reduced further and easily by employing more random forgeries in the training process.

When the Gaussian filter was not applied on the direction accumulation matrices, the AERs were substantially higher as predicted. In those experiments, the AERs for the 9 × 9 and 12 × 12 grids are 18.03% and 19.66%, respectively (See Table 5.3). As a low-pass filter, the Gaussian filter attenuates high frequency signals/information in the neighbourhood. Unlike the Gradient feature in which the Gaussian filter was employed with the intention to reduce the dimensions of the feature vector, the Gaussian Grid Feature uses the Gaussian filter to spread and preserve information. The way in which the Gaussian filter was applied to the direction code histograms in the Gradient feature causes information to spread unevenly depending on the cell’s relative position to filter centres (see Figure 5.2.7). The number of filter centres that information of a cell could transmit to also varies. These observations may help to explain the superiority of the Gaussian Grid feature over the Gradient feature despite the fact that both of them employ the Gaussian filter. It is noticed, though unintentionally, that information had also been preserved in overlapping-zone techniques such as the Flexible Grid feature [207] or the overlapping window [4]. The main difference is that, in the Gaussian Grid feature, information from the whole cell was preserved and transmitted to a larger number of surrounding zones in all directions.
Despite the encouraging results, the dimension of the feature vector of the Gaussian Grid Feature is considerably larger than other local features in the literature. In the $12 \times 12$ grid configuration, the number of feature values is 864, which is more than seven times the size of the original MDF vector. Compared to the $9 \times 9$-grid configuration, the size of the feature vector of the $12 \times 12$ configuration is nearly two times larger whilst the accuracy is only improved by 0.5%.

### 5.2.6 Analysis of Results Obtained Using the Curvature Map Feature

The results of the Curvature Map feature were not very satisfying with its large 288-dimensional feature vector. As can be seen in Table 4.24, the AER obtained was as high as 20.20%. Compared to the results employing the boundaries, which are presented in Table 4.26, the recovered trajectory slightly improved the performance of the Curvature Map feature by 0.27%. As this improvement was relatively small, hypothesis testing was performed to verify that the recovered trajectories are actually useful.

Table 5.4 presents the verification accuracies obtained using the recovered trajectories and the boundaries. In 19 out of 30 supplementary experiments, the recovered trajectories produced better results. It should be noted that the genuine and simulated forgeries employed in each supplementary experiment were identical. The statistics of these supplementary experiments are presented in Table 5.5.

**Assumptions:**

- The variable is normally distributed in both populations
- The two samples are independent
- Population variance is equal

1. **Statement of the hypothesis**
   
   $H_0 : \mu_A - \mu_B \leq 0$
   
   $H_0 : \mu_A - \mu_B > 0$

2. **Rejection rule**
Table 5.4: Verification accuracies obtained using the recovered trajectories and the boundaries

<table>
<thead>
<tr>
<th>Test</th>
<th>Recovered Trajectory (A)</th>
<th>Boundary (B)</th>
<th>A&gt;B</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0.796979</td>
<td>0.801719</td>
<td>FALSE</td>
</tr>
<tr>
<td>2</td>
<td>0.804896</td>
<td>0.799635</td>
<td>TRUE</td>
</tr>
<tr>
<td>3</td>
<td>0.810260</td>
<td>0.793646</td>
<td>TRUE</td>
</tr>
<tr>
<td>4</td>
<td>0.797865</td>
<td>0.793333</td>
<td>TRUE</td>
</tr>
<tr>
<td>5</td>
<td>0.793802</td>
<td>0.787917</td>
<td>TRUE</td>
</tr>
<tr>
<td>6</td>
<td>0.802135</td>
<td>0.795729</td>
<td>TRUE</td>
</tr>
<tr>
<td>7</td>
<td>0.797760</td>
<td>0.802083</td>
<td>FALSE</td>
</tr>
<tr>
<td>8</td>
<td>0.802292</td>
<td>0.805677</td>
<td>FALSE</td>
</tr>
<tr>
<td>9</td>
<td>0.793750</td>
<td>0.792083</td>
<td>TRUE</td>
</tr>
<tr>
<td>10</td>
<td>0.803906</td>
<td>0.800104</td>
<td>TRUE</td>
</tr>
<tr>
<td>11</td>
<td>0.792604</td>
<td>0.785990</td>
<td>TRUE</td>
</tr>
<tr>
<td>12</td>
<td>0.806875</td>
<td>0.796823</td>
<td>TRUE</td>
</tr>
<tr>
<td>13</td>
<td>0.792135</td>
<td>0.783750</td>
<td>TRUE</td>
</tr>
<tr>
<td>14</td>
<td>0.800938</td>
<td>0.793125</td>
<td>TRUE</td>
</tr>
<tr>
<td>15</td>
<td>0.796823</td>
<td>0.802448</td>
<td>FALSE</td>
</tr>
<tr>
<td>16</td>
<td>0.782604</td>
<td>0.790521</td>
<td>FALSE</td>
</tr>
<tr>
<td>17</td>
<td>0.795833</td>
<td>0.790625</td>
<td>TRUE</td>
</tr>
<tr>
<td>18</td>
<td>0.790938</td>
<td>0.787813</td>
<td>TRUE</td>
</tr>
<tr>
<td>19</td>
<td>0.795521</td>
<td>0.789271</td>
<td>TRUE</td>
</tr>
<tr>
<td>20</td>
<td>0.797500</td>
<td>0.790260</td>
<td>TRUE</td>
</tr>
<tr>
<td>21</td>
<td>0.804167</td>
<td>0.794323</td>
<td>TRUE</td>
</tr>
<tr>
<td>22</td>
<td>0.792708</td>
<td>0.795208</td>
<td>FALSE</td>
</tr>
<tr>
<td>23</td>
<td>0.800104</td>
<td>0.800781</td>
<td>FALSE</td>
</tr>
<tr>
<td>24</td>
<td>0.807240</td>
<td>0.799688</td>
<td>TRUE</td>
</tr>
<tr>
<td>25</td>
<td>0.792708</td>
<td>0.794323</td>
<td>FALSE</td>
</tr>
<tr>
<td>26</td>
<td>0.794115</td>
<td>0.789115</td>
<td>TRUE</td>
</tr>
<tr>
<td>27</td>
<td>0.797083</td>
<td>0.805365</td>
<td>FALSE</td>
</tr>
<tr>
<td>28</td>
<td>0.807188</td>
<td>0.801094</td>
<td>TRUE</td>
</tr>
<tr>
<td>29</td>
<td>0.790729</td>
<td>0.794844</td>
<td>FALSE</td>
</tr>
<tr>
<td>30</td>
<td>0.799635</td>
<td>0.800625</td>
<td>FALSE</td>
</tr>
</tbody>
</table>

Table 5.5: Statistics of results of the supplementary experiments for the Curvature Map feature

<table>
<thead>
<tr>
<th>Samples</th>
<th>A</th>
<th>B</th>
</tr>
</thead>
<tbody>
<tr>
<td>Variance ($S^2$)</td>
<td>3.7760E-05</td>
<td>3.4552E-05</td>
</tr>
<tr>
<td>Mean ($\bar{X}$)</td>
<td>0.798036458</td>
<td>0.795263889</td>
</tr>
<tr>
<td>Size (n)</td>
<td>30</td>
<td>30</td>
</tr>
</tbody>
</table>
Reject $H_0$ if

$$t_{\text{statistic}} > t_{\text{critical}(\alpha, n_A + n_B - 2)} = t(0.05, 58) = 1.672$$  \hspace{1cm} (5.2.1)

3. Test statistic computation

The pool variance

$$S_p^2 = \frac{(n_A - 1)S_A^2 + (n_B - 1)S_B^2}{n_A + n_B - 2}$$

$$= \frac{3.77601\text{E} - 05 + 3.45523\text{E} - 05}{2}$$

$$= 3.61562\text{E} - 05$$  \hspace{1cm} (5.2.2)

The standard error of the sample difference

$$S_{X_A - X_B} = \sqrt{S_p^2 \left( \frac{1}{n_A} + \frac{1}{n_B} \right)}$$

$$= \sqrt{\frac{3.61562\text{E} - 05}{15}}$$

$$= 0.001552551$$  \hspace{1cm} (5.2.3)

$$t_{\text{statistic}} = \frac{\overline{X}_A - \overline{X}_B - (\mu_A - \mu_B)}{S_{X_A - X_B}}$$

$$= \frac{0.798036458 - 0.795263889 - 0}{0.001552551}$$

$$= 1.785815712$$  \hspace{1cm} (5.2.4)

4. Decision

Reject $H_0$

5. Conclusion

At 5% level of significance there is sufficient evidence to conclude that $\mu_A > \mu_B$. In other words, the performance of the Curvature Map could be improved by employing the recovered trajectories instead of the boundaries with 95% confidence.

The above conclusion agrees with the ROC curves for the Curvature Map feature depicted in Figure 5.2.9.

![Figure 5.2.10: The performance of the local features obtained using 300dpi corpus](image-url)
CHAPTER 5. ANALYSIS AND COMPARISON OF RESULTS

Figure 5.2.9: ROC curves of the Curvature Map feature for boundary and recovered trajectory input

Figure 5.2.10 concludes Section 5.2 with the performance of local features investigated in this research.

5.3 Analysis of Results Obtained Using Global Features

The experimental results of the global features presented in Section 4.3 suggested that the use of global features could improve the performance of signature verification systems. This trend is more evident in the experiments with local features of modest performance. In this section, the results obtained from each local feature will be discussed and analysed in detail.

5.3.1 Analysis of Results Obtained Using the Ratio Feature

From the experimental results with the Ratio formulae presented in Section 4.3.1, it can be concluded that the way the extracted information is represented, it does have a certain impact on verification accuracy. The newly proposed formula (Eq. 3.3.11), which is more simple, had produced better results than Liu’s formula [132] despite both formulas having the same number of input and output values.

In the experiments using the original MDF feature, the new formula improved the AER by 0.34% from 18.36% down to 18.02%. Similarly, better results were also obtained in the experiments with the enhanced MDF on the 75dpi database (17.39% compared with 17.58%). The same trends were observed in the experiments using other local features or feature sets.

5.3.2 Analysis of Results Obtained Using the Energy Feature

As a global feature consisting of only 5 feature values, the AER of the Energy Feature is relatively high when being used separately. In the experiments using the 75dpi database, the AER obtained is 27.99%. The respective figure for the 300dpi database is 26.96%. These results indicate that the performance of the Energy feature is indeed affected by the resolution of the signature images. At 75 dpi, there are strokes whose width is inconsistent due to binarisation. Their contour sides may collapse to a single line and negatively affect the computation of the Energy feature.
CHAPTER 5. ANALYSIS AND COMPARISON OF RESULTS

17.2
17.4
17.6
17.8
18
18.2
18.4
{MDF} {MDF, R} {MDF, R2}
AER (%)
Without Energy
With Energy

Figure 5.3.1: Performance of energy-based feature for various feature combinations on the 75dpi database

Figure 5.3.2: Performance of energy-based feature for various feature combinations for the 300dpi database

Figure 5.3.1 compares the results of the original MDF-based feature sets with and without the Energy feature. It can be seen that the inclusion of Energy feature significantly enhances the performance of the MDF feature.

The experimental results presented in Section 4.3.2.3 showed that the Energy Feature had significantly improved the performance of the MDF in character recognition. It is believed that the Energy feature is an important feature of handwriting, not only in English but also other scripts, and deserves further investigation.

The Energy feature has the best performance amongst the global features investigated in this research. Figure 5.3.2 compares the performance of local features investigated in the present research with and without the this feature. In general, the inclusion of the Energy feature produces more improvement in less productive local features. The combination investigated with the Variance6x6 feature reduces the AER by 3.87%, from 22.2% down to 18.33%. A performance improvement of 0.43% was also observed in the combination with the Gradient feature. Meanwhile, the improvement for the Gaussian Grid feature was least substantial being only 0.05%, from 13.93% down to 13.88%. It is suggested that the large portion of the discriminative information conveyed by the Energy feature (global) has already been represented by the local energy information of the Gaussian Grid feature. In the experiment with the Gradient feature, there were improvements noted as the Energy feature provided the Gradient feature with new information about the relationship between the histograms of perpendicular directions.

Although the performance of the MDF and the Camastra8x8 features are comparable, the Energy feature appeared to perform better with the latter. The AER of the combination employing the Camastra8x8 feature was 16.89% whilst it was 18.47% for the MDF.
5.3.3 Analysis of Results Obtained Using the Trajectory Length Feature

The experiments indicated that the use of the Trajectory Length feature improved the results of the local features investigated. The AER of the Variance6+6 feature was improved by 1.4%. Significant improvement was also observed in the experiment using the Camastra (8x8) feature by 1.3%. However, the improvements obtained from the MDF, Gradient, Gaussian Grid, and Curvature Map features were less notable with decreases of only 0.23%, 0.08%, 0.03%, and 0.12% noted respectively. Similar to the Energy feature, the Trajectory Length feature produced better results in the experiments with the Camastra8x8 feature than with the MDF. The performance gap between these two feature combinations is 1.19%. Figure 5.3.3 presents the bar chart comparing the results of these experiments.

5.3.4 Analysis of Results Obtained Using the Moment Feature

Similar to the Trajectory Length feature, both versions of the Moment feature are rotation invariant. As can be seen in Figure 5.3.4, significant improvements were observed with the Camastra8x8 and Variance6+6 features. The AERs of the Moment3 feature in conjunction with these local features are 17.55% and 20.64%, respectively. The improvements obtained using other local features were less substantial. In every experiment with the local features, the 2-feature-values variant of the Moment feature produced better results than its single-feature counterpart. It is also noted
that despite the similar number of feature values to the MDF, the Camastra 8x8 feature produced better results in the experiments with the Moment based features.

5.4 Analysis of Results of Feature Fusion

As can be seen in Table 4.16, the performance of the Variance6+6 feature is relatively low with the best AER as high as 22.20%. However, those results were not unexpected as the Variance6+6 feature is a simple feature consisting of only 24 values. When the global features were combined with the Variance feature, the verification accuracy was improved substantially. The new AER obtained was 16.80%, which is a 5.4% improvement from the 22.20% AER. The corresponding FAR at that operational point was also reduced substantially, from 0.75% down to 0.19%. The results of this experiment are detailed in Table 4.69. These results compare favourably to the results obtained using the MDF-based feature sets previously reported in [152]. More importantly, the total number of feature values of the newly proposed feature set is only 33 compared to the dimensions of more than 120 of the MDF-based feature set. This AER of the proposed feature set is also 3.53% better than the best result of the Camastra feature.

Nevertheless, the inclusion of the set of proposed global features also significantly improves the performance of the Camastra feature by 3.96%, from 20.33% down to 16.38% AER (see tables 4.13 and 4.70). The number of feature values of this combination is 41.

5.5 Comparison to Results of Other Researchers

Apart from the investigations employing the GPDS-160 database, it is not easy to compare the performance of the proposed techniques and system with others directly. Most databases were created using different collection protocols. Signature collection protocols have a significant impact on the characteristics of a database, and subsequently the experimental results. For example, Vargas et al. [215] reported that their grey-level based feature extraction technique produced better results with the MCYT corpus compared to the GPDS. These researchers commented that the difference was due to the employment of various types of pens in the collection of the GPDS corpus. Similarly, the verification system proposed by Wen et al. [221] reported an EER of 11.4% with their proprietary corpus, which consisted of 55 signature sets, and 15.02% for the MCYT corpus. The number of genuine signatures employed for training in experiments with each database were 8 and 5 respectively. Here it should be noted that a sample selection process was employed to remove highly variant genuine signatures prior to training/testing sample separation. More recently in Batista’s research [16], the results obtained using the GPDS database were reportedly significantly worse than those from the proprietary database employed.

With respect to signature verification research employing the GPDS database, the results of the proposed techniques compare favourably to the results reported by Batista [16], Yilmaz et al. [225], and Kumar et al. [115]. In these papers, the researchers did not use simulated forgeries to train their classifiers. The AERs obtained using the GPDS database in Batista's work were well over 20% when 12 genuine signatures were employed for training. Yilmaz et al. employed weighted sum rules to combine the results of the proposed features. Despite their vague methodology, these researchers claimed an AER as low as 15.41%. In Kumar et al.’s work, the researchers claimed an EER of 13.76% with a writer-independent approach. Unlike the present research in which the simulated forgeries were employed to tune the common parameters of the classifiers, Kumar
et al. employed the genuine-simulated forgery distances to train their single classifier. The one half of the signature corpus was employed to compute those distances whilst the other half was used for testing. This approach supposedly provided the classifier with statistical information about the genuine-genuine and genuine-forgery distances. However, it is unclear that whether the adoption of a single classifier for every signer is appropriate as the distribution of such distances may vary significantly amongst signers. Unfortunately, the current research failed to replicate their experimental results.
Chapter 6

CONCLUSIONS AND FUTURE RESEARCH

The research presented in this dissertation has investigated local and global feature extraction techniques for the problem of off-line signature verification. Its main objectives were to investigate and improve the performance of the Modified Direction Feature (MDF) and to create better feature extraction techniques. Both these objectives have been successfully accomplished.

The experimental results obtained clearly indicate that the newly proposed averaging scheme has improved the performance of the MDF substantially when the 75dpi database was employed. Specifically, an AER of 17.58% was obtained using the 9-group configuration. In the experiments using the 300dpi database, better results could also be achieved by increasing the number of groups with the trade-off of increasing feature vector dimensions.

Apart from the MDF, additional state-of-the-art local features proposed by Camastra [32] and Wakabayashi et al. (Gradient features) [220] have also been investigated. Although the performance of the MDF was not as good as the Camastra feature for the handwritten character recognition problem [208], the MDF produced better results in signature verification. Nevertheless, both the MDF and Camastra features had been outperformed by the Gradient feature, which possesses a much larger 576-dimensional feature vector. These findings were published in the ICFHR 2010 conference [154]. Performance analyses of these local features have set the foundation for the development of the Gaussian Grid, a novel grid-based feature extraction technique.

Despite its simplicity in computation, the performance of the Gaussian Grid feature is comparable or significantly better than other state-of-the-art techniques. This was achieved by appropriately applying the 2D Gaussian filter on a set of six matrices of directional information. More importantly, the Gaussian filter can be easily applied to other zone-based feature extraction techniques in which the accumulation of information occurs. This is an advantage of information preservation using a Gaussian filter over blurring input images directly. The investigation of the Gaussian Grid feature presented in this dissertation has been published in the ICDAR 2011 conference [155]. Employing the identical optimal set of parameters, results of the experiments with a proprietary Bangla signature corpus [159] also confirm the robustness of Gaussian Grid feature.

The second, newly-proposed grid-based local feature in this research was the Curvature Map feature. This feature provides the classifiers with information about the geometric distribution of high curvature sections of the signature by utilising the information produced by the newly-
CHAPTER 6. CONCLUSIONS AND FUTURE RESEARCH

proposed trajectory recovery technique. Although the performance of the Curvature Map was not as high as expected, the results obtained suggested that features extracted from the recovered trajectory can be useful and the application of the recovered trajectory and the proposed technique are worth further investigation. To the best of the author’s knowledge, the Curvature Map feature is the first ever investigated feature extraction technique that utilises the recovered trajectory for feature analysis.

The Variance feature is the third newly proposed local feature of this research. Despite its modest performance compared to other local features investigated in this research, the Variance feature has the smallest dimensions of all. In its 6-group setting, the dimension of the feature vector is only 24. Its small size and reasonable performance paved the way for the investigation of compact feature sets using global features.

Unlike their global counterparts, local feature extraction techniques have been investigated intensively in the area of off-line signature verification. In many cases, the dimensions of the feature vectors are large, ranging from hundreds to thousands. Large sized features tend to create a heavy load on the classifiers employed. Besides, they can diminish the contribution of “good” but small “add-on” features. The present research has demonstrated that competitive results could be obtained using a relatively small-dimensional feature set by employing global features. The performance of the Variance6+6 in conjunction with other global features compares favourably to the MDF and Camastra features in terms of overall performance as well as feature vector size.

Amongst the four global features investigated in this research, the Energy feature has the highest distinguishing power. This feature further reduced the AER of the Variance6+6 and Camastra8x8 local features by 3.87% and 2.47%, respectively. The performance of Moment3, Ratio, and Trajectory Length were not as good as the Energy feature due to their smaller feature vectors. Nevertheless, the combination of all global features produced the best performance. In the experiments using the Variance6+6 and the Camastra features, the improvements were 5.4% and 2.97%, respectively.

6.1 Future Research

A large number of experiments have been conducted in the current research. As a result of this, numerous solid experimental results and useful observations have been demonstrated, and future research may investigate this area further in order to improve the performance of off-line signature verification systems.

The performance of the Gaussian Grid feature is affected by the quality of the input contour, which again heavily relies on the preprocessing techniques employed. This is due to the use of signature contours as inputs. It is suggested that more robust directional information extraction techniques should be employed instead of relying solely on the contours. Apart from that, supplementary manipulation on the accumulated direction data matrices should be investigated in order to provide the classifier with additional information.

It is observed that the inclusion of global features produces more performance improvement with the Camastra8x8 feature than with the MDF. The differences are 1.77%, 1.19%, and 1.58% for the Moment3, Trajectory Length, and Energy features, respectively. It is also noted that the MDF and the Camastra8x8 features have similar performance. The number of feature values of the MDF and Camastra8x8 feature are also approximate. These similarities need to be investigated further as it may give an insight into the designing of better feature extraction technique for signature
verification.

Despite the encouraging results of the \{Variance6+6, Ratio, Energy, Trajectory Length, Moment3\} feature set, the difference in performance compared to other state-of-the-art local feature extraction techniques is still large. However, it is strongly believed that they could be bridged by employing additional global features, especially those that are rotation invariant. Fusion techniques of global features in off-line signature verification should also be investigated, as it has reportedly produced encouraging results [64] in the on-line counterpart. The literature reviewed also indicated that there hasn’t been any global feature extraction technique that is capable of creating an arbitrary number of feature values. Future research in this direction may extend the Moment-based feature using concentric circles.

With respect to the Intersection Analysis framework, geometric fitting algorithms for other curves could be employed for better bridging score evaluations and more reliable segment matching. It is strongly posited that this direction of research would help to increase the overall accuracy of the recovered contour.
Bibliography


[188] Laetitia Rousseau, Jean Camillerapp, and Eric Anquetil. What knowledge about handwritten letters can be used to recover their drawing order. In 10th IWFHR, pages 355–360, La Baule, France, Oct 2006. IRISA/INSA - Campus Universitaire de Beaulieu.


